特别说明

此资料来自百度文库（http://wenku.baidu.com/）

您目前所看到的文档是使用的**抱米花百度文库下载器**所生成

此文档原地址来自

http://wenku.baidu.com/view/3b810e4f2b160b4e767fcfe4.html

感谢您的支持

抱米花

http://blog.sina.com.cn/lotusbaob

VISIT…

LANZAROTE Caliente.com

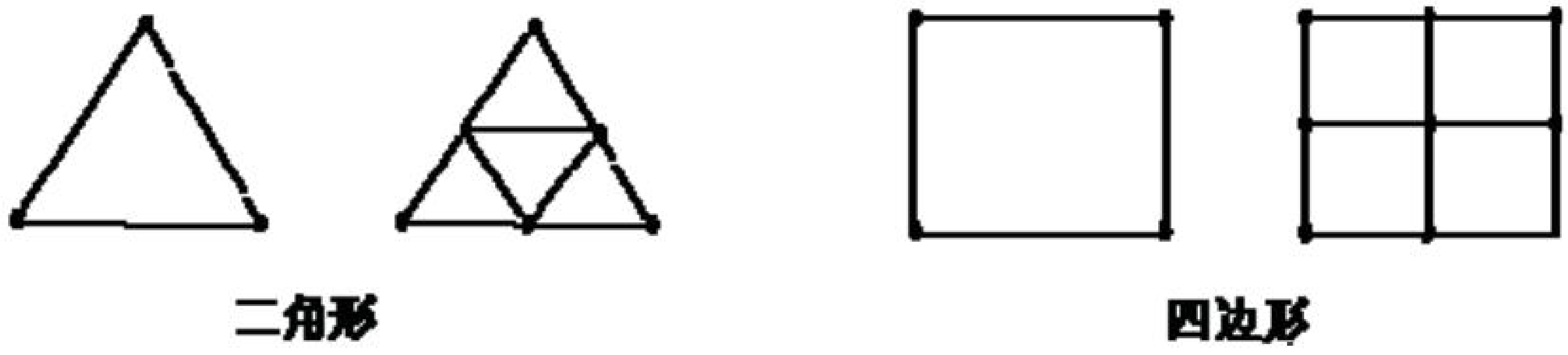


图 5.2.5：2D 单元类型的悬挂节点适应

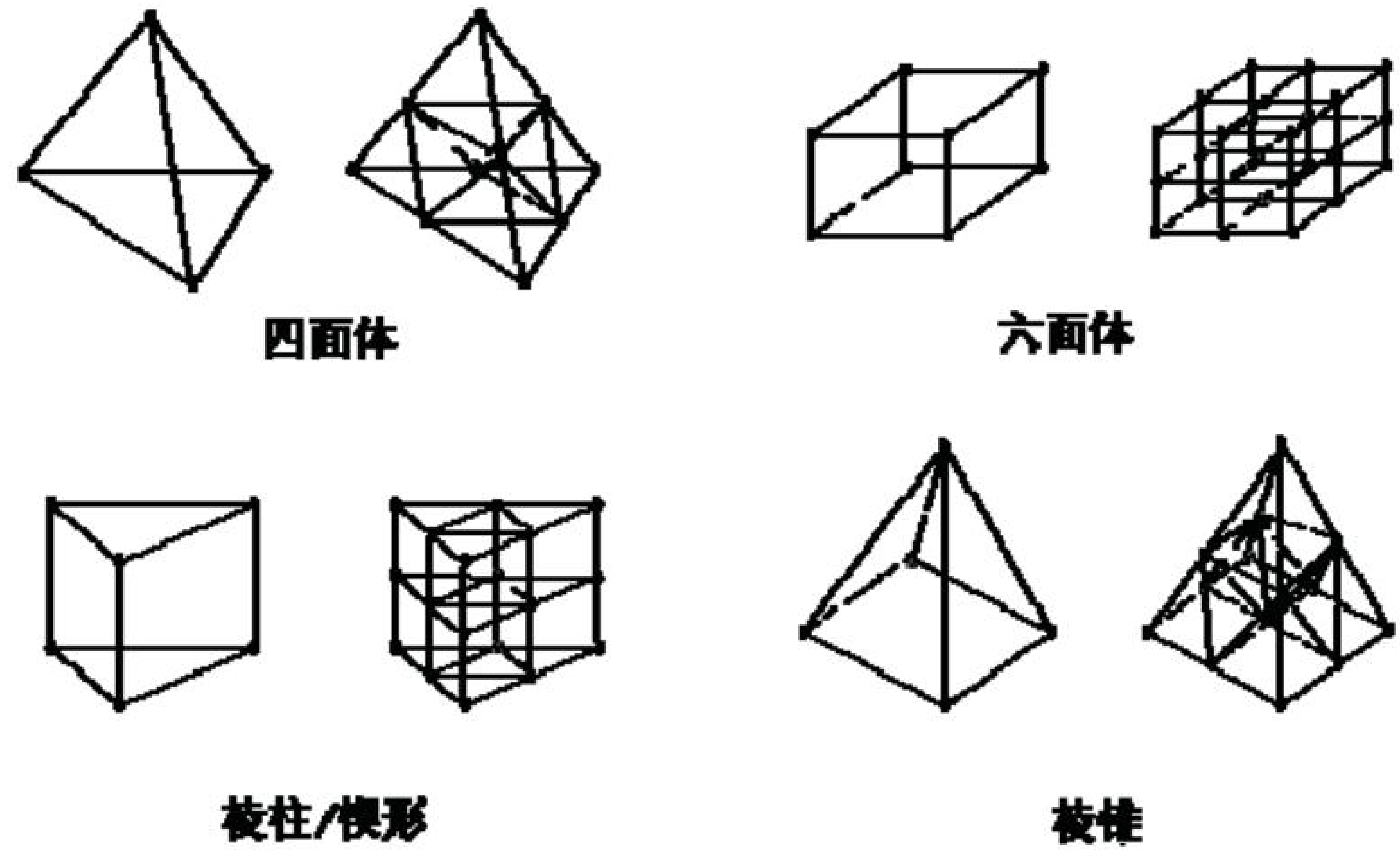


图 5.2.6：3D 单元类型的悬挂节点适应

悬挂节点的粗化

单元是通过再引入不活动的父单元被粗化的，即结合子单元以利用前面被细分的父单元。如果所有的子单元被标记为粗化，那么不活动的父单元被恢复活动。你最终将利用最初的网格，其中这个网格的悬挂节点粗化被重复应用。你不能使用悬挂节点适应方法比最初的网格更进一步的粗化网格。然而，粗化允许你移动最初的网格点以减少网格密度。

5.2.3 正则适应

正则适应过程不创建悬挂节点，而是共享一个边或面的所有单元都包含那些实体上的所有节点。注意，这个程序不适合与动态适应联合使用。

正则细化

要细化单元，你可以把边界或内部面（包括周期边界面）切开。图 5.2.7 给出了标签为 A 的三角形怎样被切开而细化的。单元通过分三角形或四面体的最长边被细分的。这一技术有两个好处：过程保守，不要求用插补来获得新单元的解矢量，另一方面，重复细分一个歪斜的单元不会继续增加网格的偏斜。

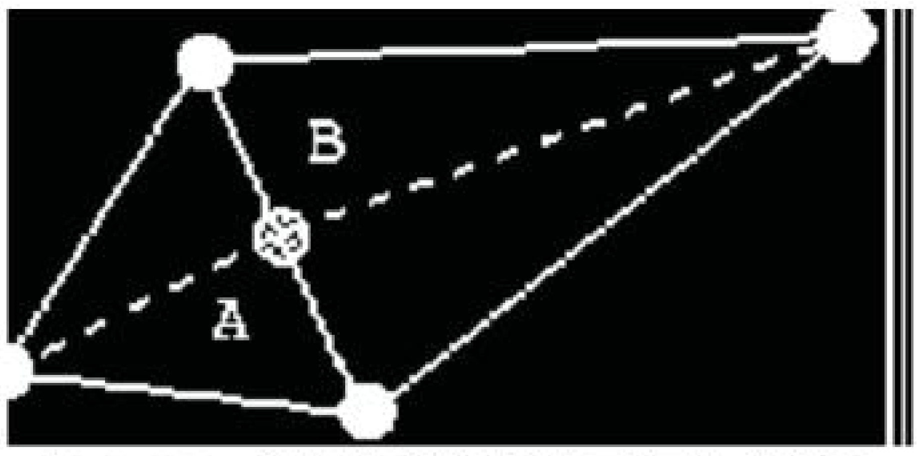


图 5.2.7：通过切分最长边而细化的单元

现在的这个方案假设已找到了标记为细化的任何单元的最长边。这个方案访问每个包含这样边的单元，并且寻找一个比较长的边。一旦它找到了最长的边，就会切开这条边。虽然这个程序保持了重复分成三角形的性质，但它也造成了许多没有标记为细化的单元被切开的现象。例如，图 5.2.8 给出了原来标记为细化的单元（标记为 X），图 5.2.9 给出了通过细化程序而建立的最终单元。

正则粗化

网格通过删除标记为粗化的单元上的节点而被粗化。如果依附在节点上的所有单元被标记为粗化，那么求解器企图删除节点。企图从标记为切除的每个节点出发而局部重分三角形的过程如下：

一列依附在标记节点上的单元被生成。切除这些单元将来一个必须被重新分成三角形的腔。

腔内的一列面被生成。

腔的边界上的一列面被生成。

如果要切除的节点在一个边界上，那么一个新边界的重分三角形被生成，及那些面被添加到腔上的这列面中。

从腔上的这列面出发，一个新的 Delaunay 重分三角形被建立。

如果程序成功，区域中原始三角形上的节点、面和单元被删除。

所有与腔相关的节点被从这列命定的节点中切除以避免在同一区域中产生连续的粗化。

在新单元中的解变量是使用一个体积的加权平均计算的。

图 5.2.10 给出了节点 n1 的切除和区域被重新分成三角形的结果。在这个例子中，附在节点上的这列单元包括 c1，c2，c3，c4 和 c5；腔内的这列面包括 f6，f7，f8，f9 和 f10；

腔上的这列面包括 f1，f2，f3，f4 和 f5。重分三角形的新面是 f11 和 f12，新单元是 c6，c7 和 c8。

由细化所引入的节点被称为细化节点。网格中的节点在细化之前被称为最初（或原始）节点。缺省时，仅有细化节点可以在粗化过程中被切除，但是你可以通过重新设置节点标志而切除任何节点。关于节点标志的内容在第 5.10 节中有详细介绍。

目前，网格粗化的方案只可以在 FLUENT 的 2D 版本中执行。

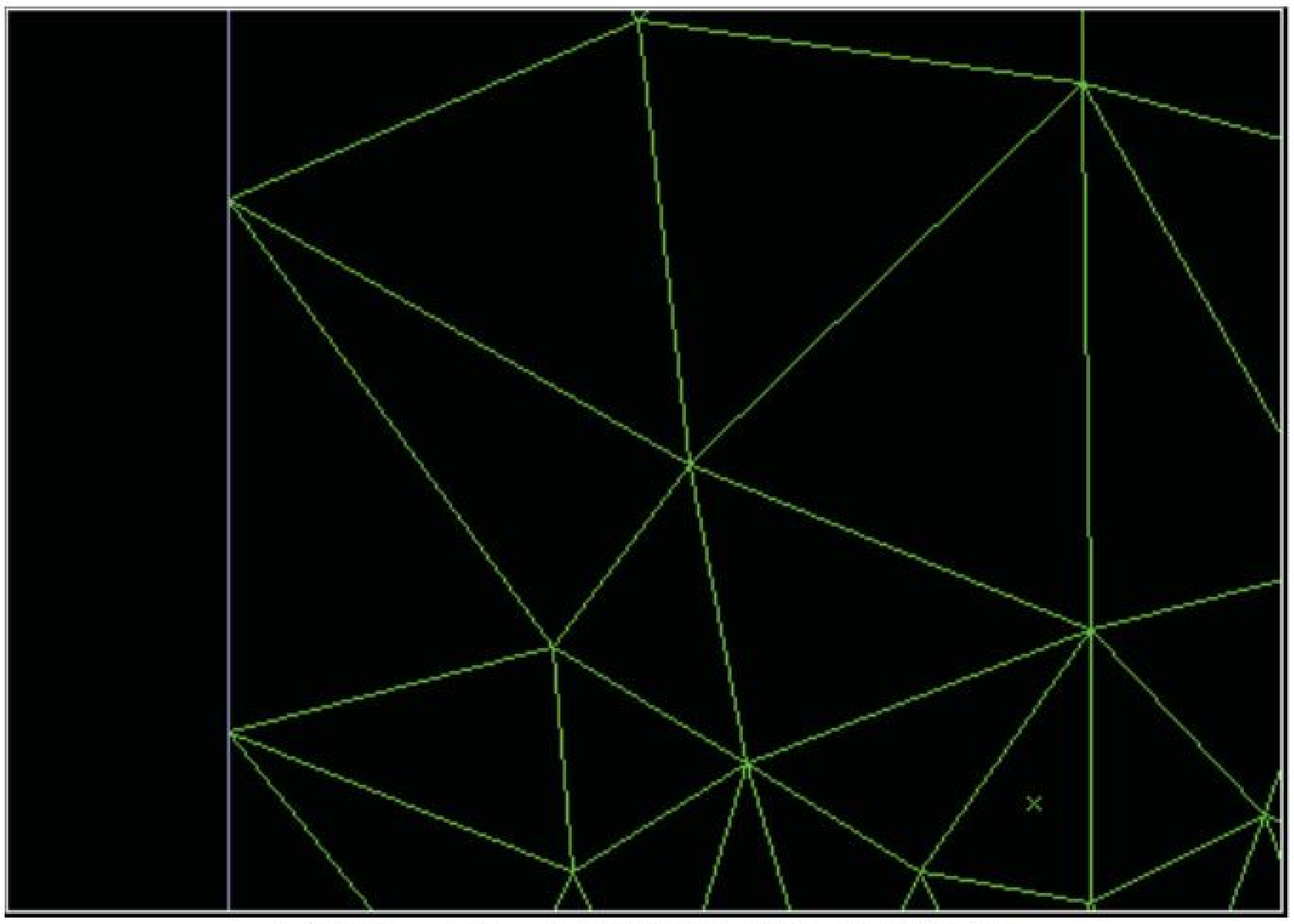


图 5.2.8：包含一个标记为细化的单元的原始网格

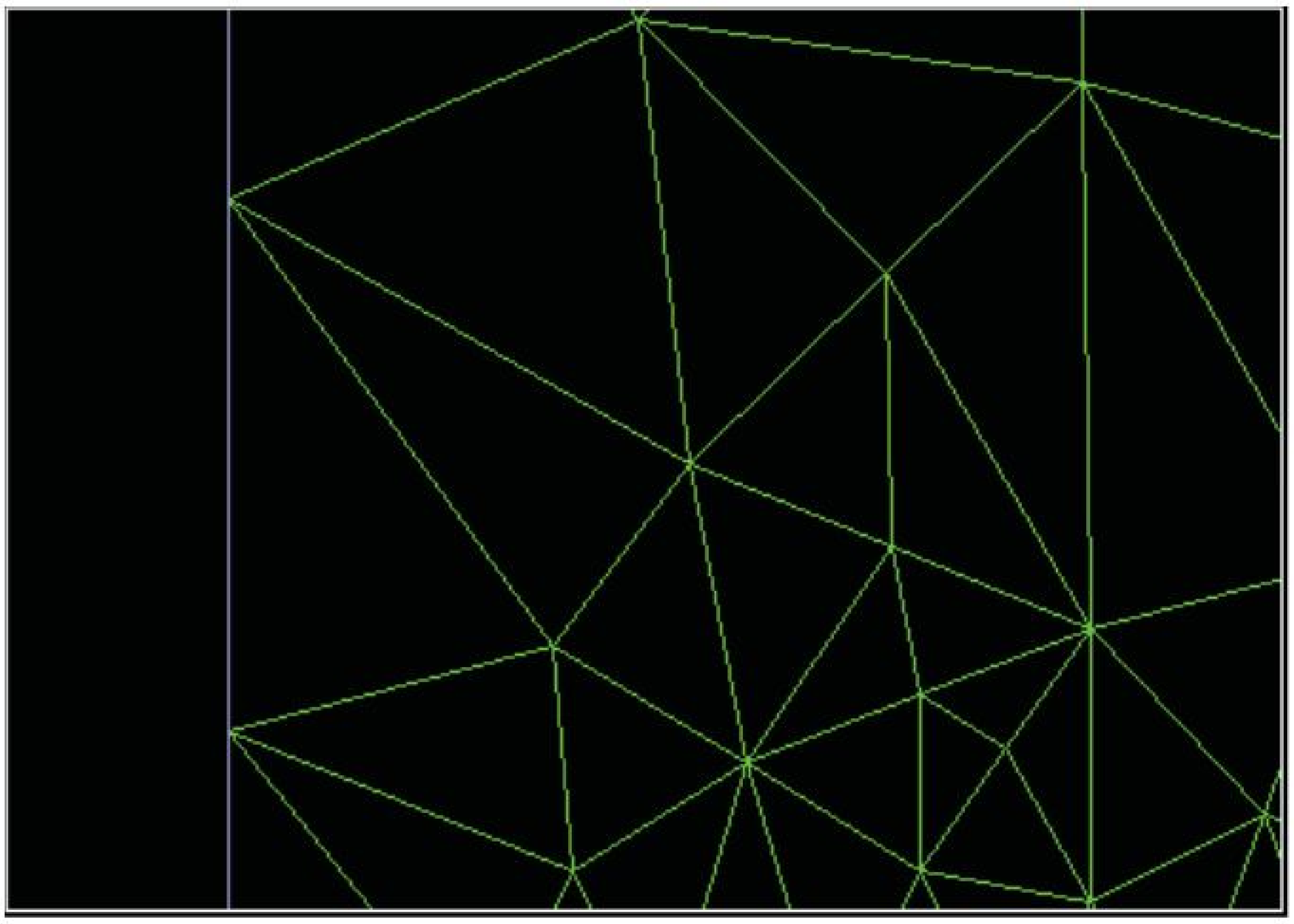


图 5.2.9：细化程序之后的最终网格

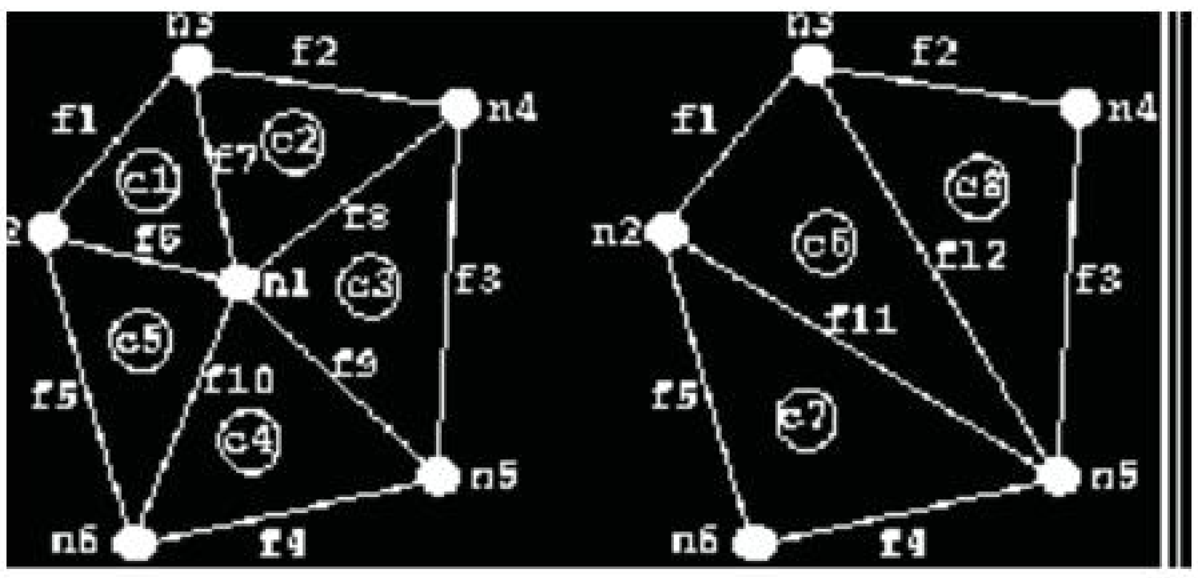


图 5.2.10：通过切除一个节点及对域重分三角形而粗化网格

5.2.4　正则与悬挂节点的适应性对比

对于大多数问题，悬挂节点的适应性为网格适应提供了相当大的机动性。然而，下面几点将会帮助你针对特殊的应用选择适当类型的适应性。

正则适应性的方法仅适用于三角形和四面体网格，而悬挂节点适应性则可以应用于所有被支持的单元类型。

悬挂节点适应性通常在本质上比正则适应性更具有局部性。在正则适应中，除了被标记的单元，许多单元可以由最长的边分开标准而被细化。对于高级别的网格，原始的正则细化延伸趋于展示单元细化的真实传播，有时细分许多远离标记为细化的实际单元的网格单元。悬挂节点方案仅宣传维持细化水平的差异，所以比较有局限性。

原始网格的连通性在悬挂节点适应性方案中仍保持，但正则适应性方法将修改关于细

化和粗化适应的连通性。对于网格被用于一个有周期行为的非稳定问题（例如，一个圆柱体后面的涡脱落），如果你执行连续的细化和粗化，这可能牵连着精确性问题。然而，仅有正则粗化允许你粗化原始的网格，并且这仅适用于 2D 问题。

悬挂节点适应性有一个与维持网格层次和临时存储 3D 中的边相联系的内存惩罚问题。正则适应性除了附加的节点、面和单元被添加以增加网格密度外则没有内存费用问题。

正则适应性不应当用于动态适应性。

5.3　边界适应

如果一个边界上需要更多的单元，那么可以使用边界适应来添加单元。边界适应函数允许你在邻近所选择的边界域内对单元进行标记或细化。之所以 FLUENT 有对邻近一个或多个边界域的单元进行细化的能力，是因为重要的流体交互作用经常发生在这些区域，比如强速度梯度在接近一个壁面的边界层内发展。

5.3.1　边界适应的例子

图 5.3.1 给出了一个关于网格可以使用边界适应而改进的例子。这个网格在台阶的竖直面上仅有两个单元。在与这个竖直面相应的区域上，边界适应可以被用来增加单元的数目，如图 5.3.2 所示。注意，这个步骤不被增加一个弯曲表面的分解。所以，如果在一个比较重要形状的曲面上需要更多的单元，最好在把网格读进求解器之前建立这个含有重要表面节点的网格。

5.3.2　执行边界适应的步骤

三种不同的方法可以用于边界适应：

基于一个单元到边界的距离的适应，在数单元中被测量

基于一个单元到边界的正则距离的适应

基于一个目标边界体积和增长因子的适应

你可以在 **Boundary Adaption** 面板（如图 5.3.3 所示）中使用这三种方法的任一个。

A → Bounda

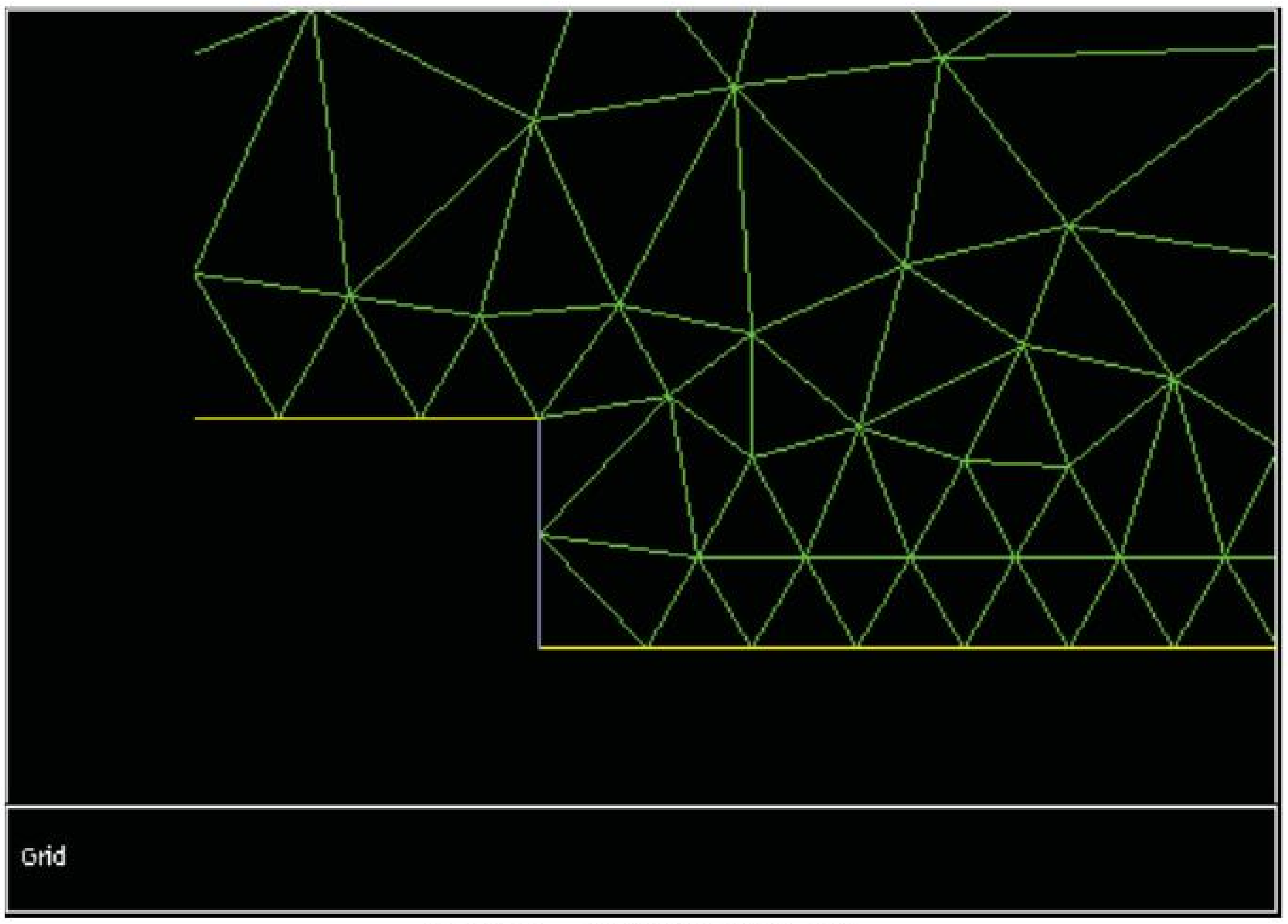


图 5.3.1：边界适应前的网格

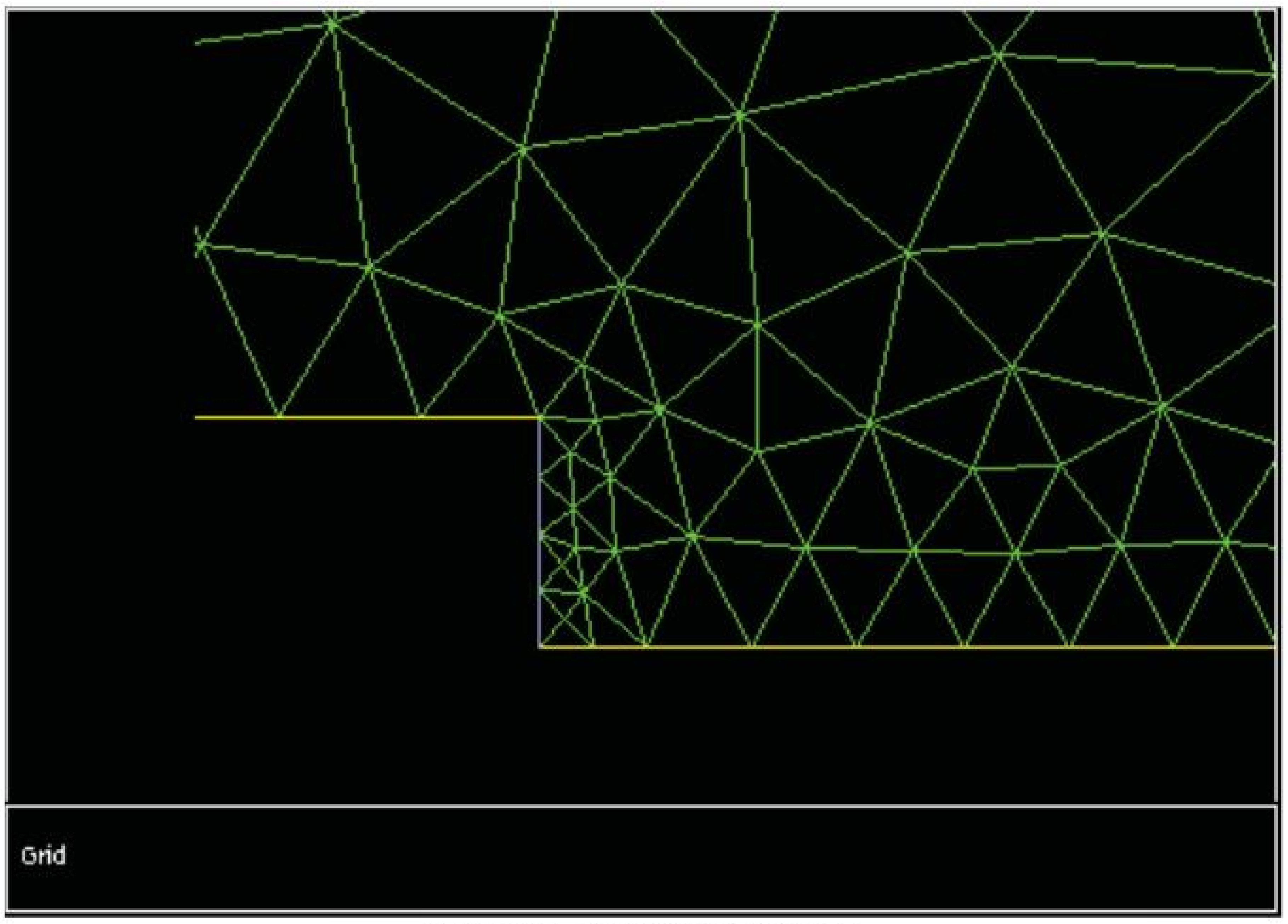


图 5.3.2：边界适应后的网格

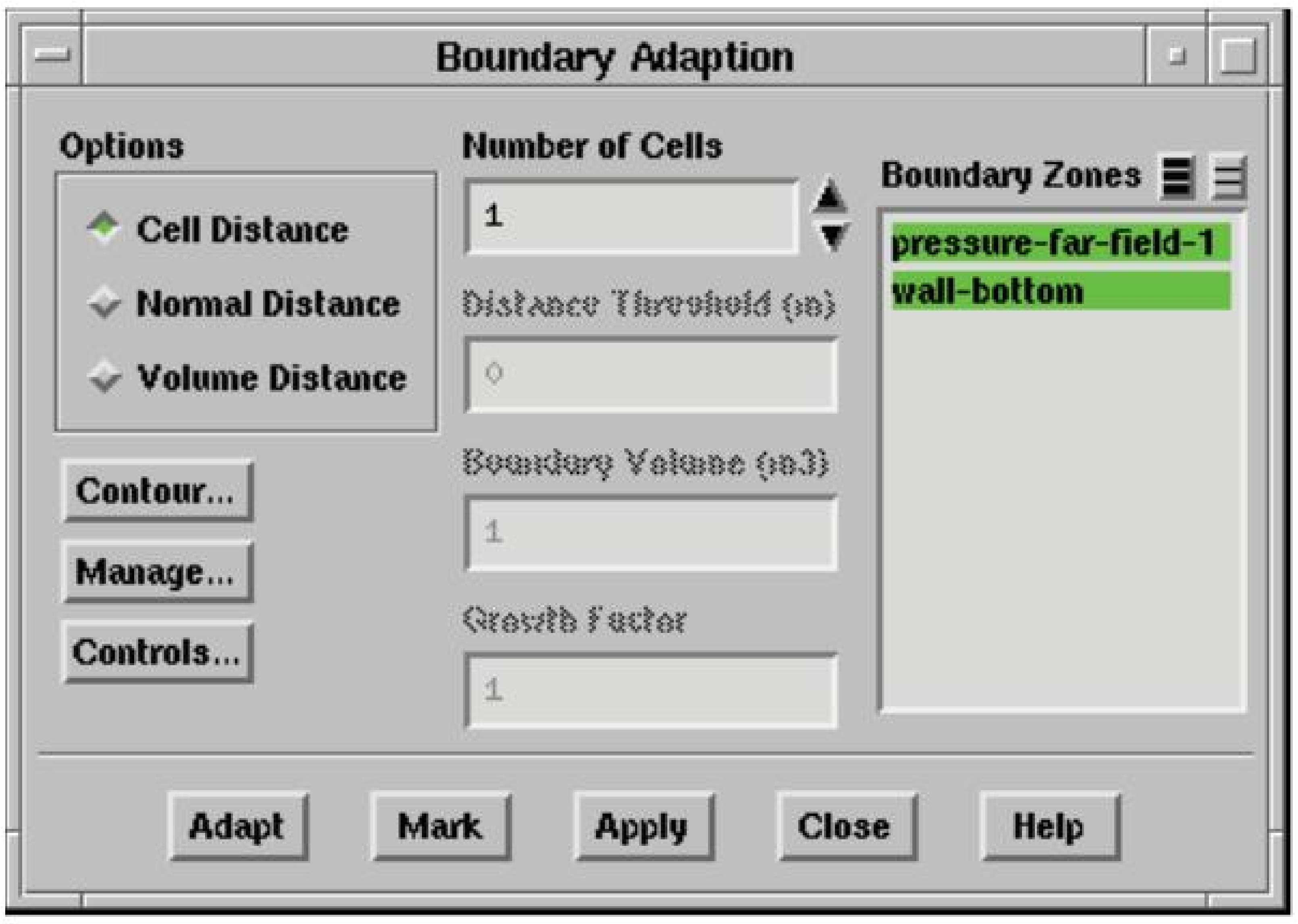


图 5.3.3：Boundary Adaption 面板

基于单元个数的边界适应

从单元的个数出发，在一个单元到边界的距离的基础上执行适应的一般步骤如下：

1. 在 Boundary Adaption 面板上，选择 Options 下面的 Cell Distance，在 Boundary Zones 列中选择你想细化的单元所邻近的边界域，然后点击 Apply。

2．点击 Contour…按钮打开 Coutours 面板。

3．在 **Contours** 面板中，激活 **Filled**，使 **Node Values** 不活动，在 **Contours Of** 下面的列中选择 **Adaption…**和 **Boundary Cell Distance**，选择合适的表面（仅对 3D），点击 **Display** 查看单元的位置，其中每个单元都有边界单元距离值。通过展示值的不同范围，你可以决定你想进行适应的单元的单元距离。

4．在 **Boundary Adaption** 面板中，设置 **Number of Cells**。如果你保留缺省值 1，那么仅有那些在指定的边界域上有边（2D）或面（3D）的单元（即一个边界单元距离为 1 的那些单元）将被标记或适应。如果把值增加到 2，一个边界单元距离为 2 的单元也将被标记/适应，依此类推。

5.（可选择的）如果你想设置任何适应选项（在第 5.10 节有介绍），点击 **Controls…** 按钮打开 **Grid Adaption Controls** 面板。

6. 点击 **Mark**，这样通过把单元放进一个适应记录而标记单元为细化，或着点击 **Adapt** 立即执行细化。

基于正则距离的边界适应

关于执行基于一个单元到边界的正则距离的细化的一般步骤如下：

1. 在 Boundary Adaption 面板，选择 Options 下面的 Normal Distance，在 Boundary Zones

列中选择你想细化的单元所邻近的边界域，然后点击 Apply。

2．点击 Contour…按钮打开 Contours 面板。

3．在 **Contours** 面板中，激活 **Filled**，使 **Node Values** 不活动，在 **Contours Of** 下面的列中选择 **Adaption…**和 **Boundary Normal Distance**，选择合适的表面（仅对 3D），然后点击 **Display** 查看单元的位置，其中每个单元都有正则距离值。通过展示值的不同范围，你可以决定你想进行适应的单元的正则距离。

4．在 **Boundary Adaption** 面板中，设置你所需要 **Distance Threshold** 值。到所选择的边界区域的一个正则距离小于或等于这个值的单元将被标记或适应。

5．（可选择的）如果你想设置任何适应选项（在第 5.10 节有介绍），点击 **Controls…** 按钮打开 **Grid Adaption Controls** 面板。

6. 点击 **Mark**，这样通过把单元放进一个适应记录而标记单元为细化，或着点击 **Adapt** 立即执行细化。

基于目标边界体积的边界适应

这个边界适应允许你在一个目标边界单元体积和指数增长函数的基础上细分单元。单元基于下面的方程被标记为细化：

$V_n > V_{boundary} e^{\alpha d}$ （5.3.1）

其中，V_n 是单元的体积，$V_{boundaary}$ 是规定的边界体积（Boundary Volume），α 是指示增长因子（Growth Factor），d 是单元质心到所选边界的正则距离。$V_{boundary} e^{\alpha d}$ 是一个单元的目标体积。

关于这种类型的边界细化的一般步骤如下：

1．在 Boundary Adaption 面板，选择 Options 下面的 Volume Distance，设置 Boundary Valume 和 Growth Factor 的值，在 Boundary Zones 列中选择你想使用的边界区域，然后点击 Apply。

2．点击 Contour…按钮打开 Contours 面板。

3．在 Contours 面板中，激活 Filled，使 Node Values 不活动，在 Contours Of 下面的列中选择 Adaption…和 Boundary Normal Distance，选择合适的表面（仅对 3D），然后点击 Display 查看目标体积的等值线。你可以修改任何输入的值（Boundary Volume，Growth Factor，和 Boundary Zones），在 Boundary Adaption 面板中点击 Apply，然后再展示等值线图以显现修改的目标体积分布。

4．（可选择的）如果你想设置任何适应选项（在第 5.10 节有介绍），点击 **Controls…** 按钮打开 **Grid Adaption Controls** 面板。

5. 点击 **Mark**，这样通过把单元放进一个适应记录而标记单元为细化，或着点击 **Adapt** 立即执行细化。

5.4 梯度适应

梯度适应函数允许你在所选择的场变量的梯度（或曲率）基础上标记单元或适应网格。

5.4.1 梯度适应途径

解适应的网格细化的基本目的是有效地降低数值解中的数值错误，最小化数值成本。可惜的是，由于在适应网格中正确地估计和对误差建模都比较复杂，因此关于点插入适应方案的直接误差估计是困难的。事实上，关于误差估计和收敛，在数学上还没有全面严格的理论可以用于 CFD 模拟。假设最大的误差发生在高梯度区域，展开流场是容易用的物理特性，它可以用来推进网格适应的进程。对网格适应来说，使用这个信息在 FLUENT 中有两个途径：

第一个梯度途径建议在强冲击的问题中使用，例如，超音速无粘流动。在这个途径中，一个特征长度乘上所选解变量的梯度的未划分的 Euclidean 标准。例如，两维中的梯度函数的形式如下：

$$|e_{i1}| = (A_{cell})^{\frac{r}{2}}|\nabla f| \quad (5.4.1)$$

这里，e_{i1} 是误差指标，A_{cell} 是单元的面积，r 梯度体积重量，∇f 是场变量 f 的梯度的未分割的 Euclidean 标准。梯度体积重量的缺省值是统一的，它符合满体积加权；一个 0 值将消除体积加权，0 到 1 之间的值将使用体积的比例加权。

第二个梯度（曲率）途径建议用于流畅的解问题。这个是 FLUENT 原来使用的等分布适应技术，一个特征长度乘上所选解变量的未分割的 Laplacian 算子。例如，二维中梯度函数的形式如下：

$$|e_{i2}| = (A_{cell})^{\frac{r}{2}}|\nabla^2 f| \quad (5.4.2)$$

其中，e_{i2} 是误差指标，A_{cell} 是单元的面积，r 梯度体积重量，$\nabla^2 f$ 是场变量 f 的未分割的 Laplacian 算子。梯度体积重量的缺省值是统一的，它符合满体积加权；一个 0 值将消除体积加权，0 到 1 之间的值将使用体积的比例加权。

长度是单元体积的平方根（2D）或立方根（3D）。长度的引入均允许强和弱干扰的分解，从而为获得更精确的解增加潜力。然而，你可以通过在 **Grid Adaption Controls** 面板中改变梯度 **Volume Weight** 而减少或消除体积加权。

任何适用于做等值线的场变量都可以用于梯度适应函数。有趣的是，这些刻度函数包括数值解的几何和物理两种特性。所以，除了对物体特性的传统适应，例如速度，你可以选择对单元体积场进行适应以减少单元体积内的快速变化。

除了 FLUENT 原来用的 **Standard**（非规格化）途径，下面两个选项适应于 **Normalization**：

1．**Scale**，它是 e_{i1} 或 e_{i2} 的值与其在区域内的平均值的比例，即：

$$\frac{|e_i|}{|\bar{e}_i|} \quad (5.4.3)$$

当使用 **Scale** 选项，关于 **Coarsen Threshold** 和 **Refine Threshold** 的适当的首切值分别是 0.3 到 0.5，和 0.7 到 0.9。较小的值将导致较大的适应的区域。

2．**Normalize**，它是 e_{i1} 或 e_{i2} 的值与其在区域内的最大值的比例，即：

$$\frac{|e_i|}{\max|e_i|} \quad (5.4.4)$$

当使用 **Normalize** 选项时，关于 **Coarsen Threshold** 和 **Refine Threshold** 的适当的首切值分别是 0.2 到 0.4，和 0.5 到 0.9。较小的值将导致较大的适应的区域。

5.4.2　稳定的梯度适应例子

越过一个循环圆柱体的超音速流动的解是使用梯度适应的一个比较好的例子。初始网格，如图 5.4.1 所示，即使包含了足够能描述圆柱体形状的网格，但仍很粗糙。圆柱体前头的网格太粗糙以致不能解决形成于圆柱体前面的冲击波。在这个例子中，在经过冲击的压力中将有一个跳跃，所有显然压力适合用于梯度适应。然而，在冲击能够被适当解决之前，必须有几个适应。几个适应之后的网格如图 5.4.2 所示。

关于一个不可压缩流动的梯度适应的一个有代表的应用可能是一个涉及一中断的混合层。

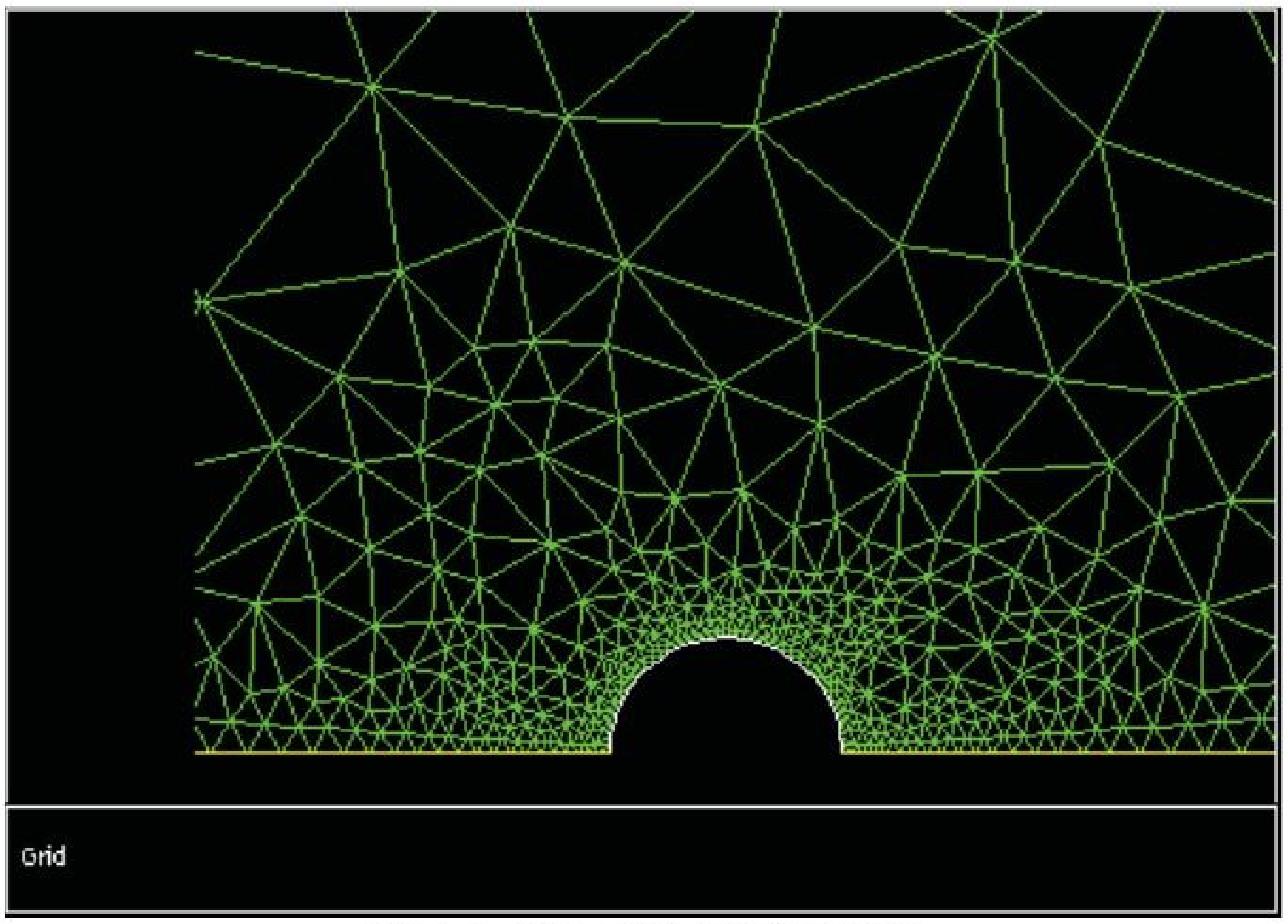


图 5.4.1：适应之前的非流线型体网格

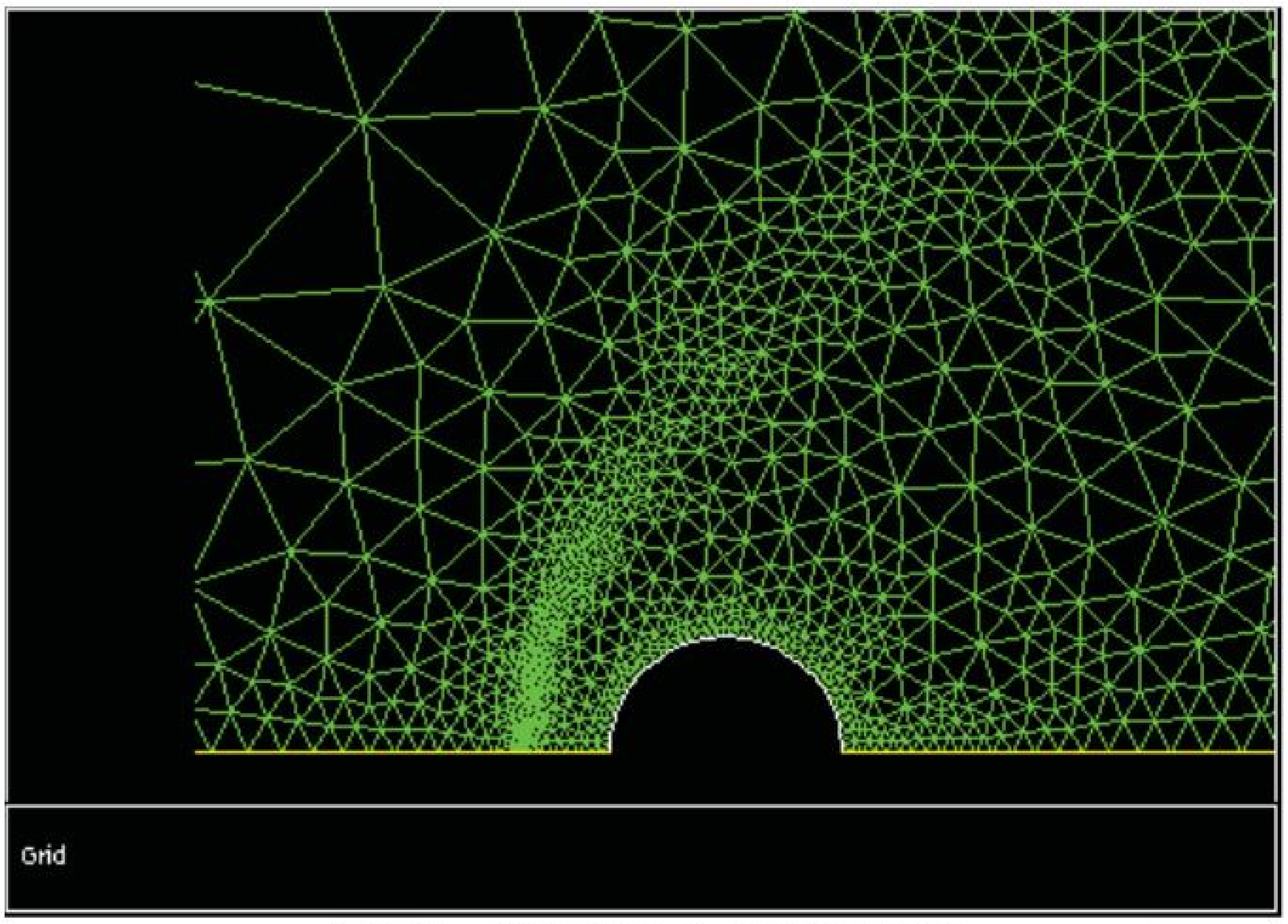


图 5.4.2：梯度适应之后的非流线型体网格

5.4.3　执行梯度适应的步骤

在 **Gradient Adaption** 面板（如图 5.4.3 所示）中执行梯度适应。

A

Gradien

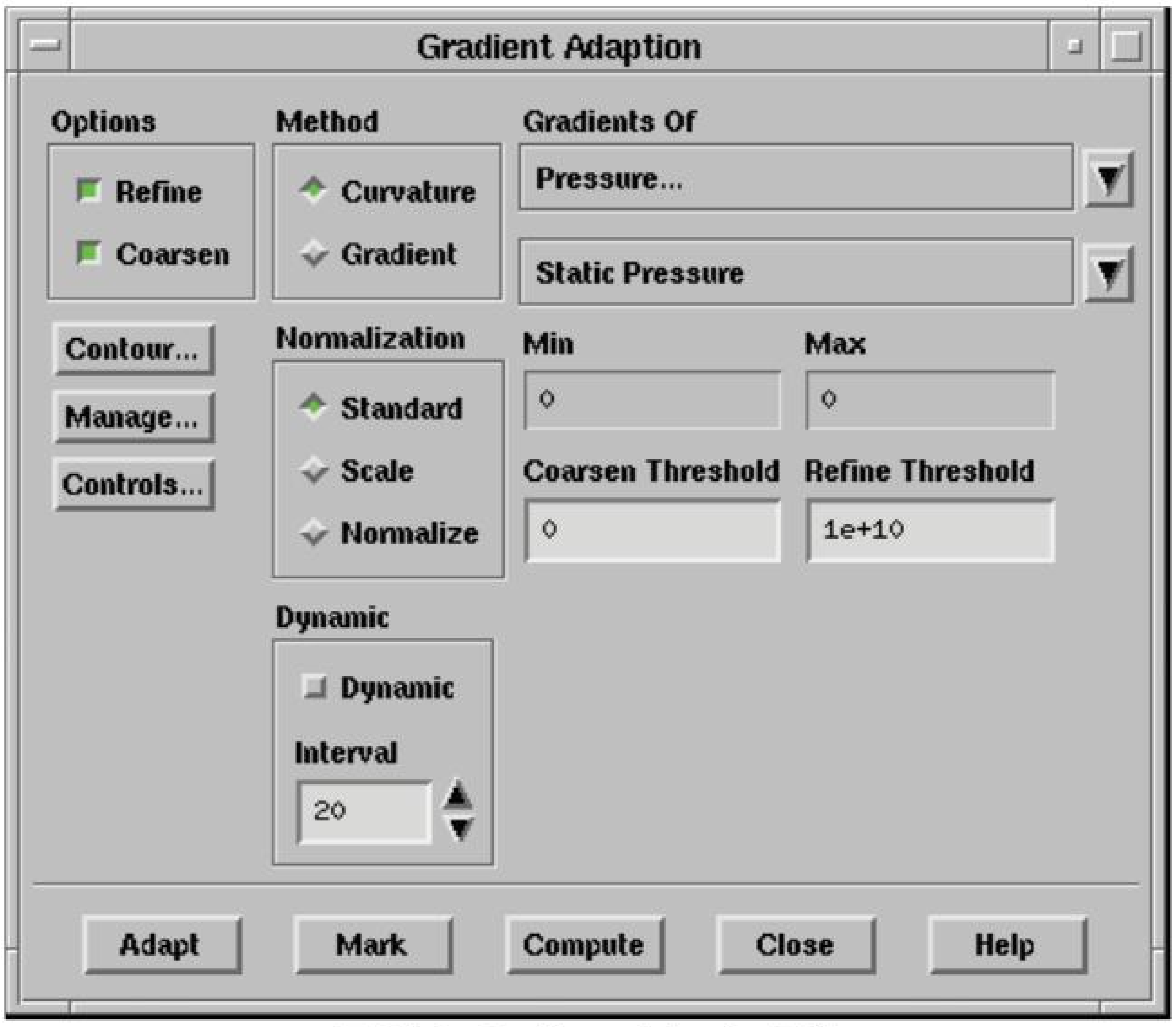


图 5.4.3：Gradient　Adaption 面板

执行梯度适应的一般步骤如下：

1．在 Grandient Adaption 面板中，选择 Gradients Of 下面的列中所需要的解变量，然后点击 Compute。

2．点击 Contour…按钮打开 Contours 面板。

3．在 **Contours** 面板中，激活 **Filled** 等值线，使 **Node Values** 不活动，在 **Contours Of** 下面的列中选择 **Adaption…**和 **Existing Value**，选择合适的表面(仅对 3D)，然后点击 **Display** 查看单元的位置，其中每个单元都有梯度值。通过展示值的不同范围，你可以决定你想使适应的单元的梯度范围。注意，如果你使用规格化，那么任何变量的梯度的范围将保持在[0, 1]。

4．在 **Gradient Adaption** 面板中设置 **Refine Threshold**。有梯度值的单元将被标记或细分。在 **Gradient Adaption** 面板中，你也会发现类 **Method** 和 **Normalization**。Curvature 是关于 FLUENT 原来所使用的适应的标准；Grandient 是一个比较新的标准，为了适应性，它使用一个场变量的第一个梯度，而不是第二个梯度（曲率）。取决于所求解的问题，你可以使用一个或其它的适应方法：

Curvature 被建议用于流畅解的问题。

Gradient 被建议用于强冲击的问题，例如，超音速无粘流动。

进一步的选项还有：

Standard 意味着没有规格化的梯度或曲率被执行。这中情形一直存在于 Fluent 6.0 中。

Scale 意味着梯度或曲率是对区域中的平均值的比例。

Normalize 使用一个对域中变量的最大值的一个缩放比例，即梯度或曲率的取值范围为[0, 1]。

使用缩放比例或规格化使细化和粗化临界值的设置变地简单了，并且几乎独立于当前的解和特定问题。这对于使用自动的动态程序尤其重要。

5．如果你想粗化网格，设置 Coarsen Threwshold 为非零值。带有梯度值（在特定值下面）的单元将被标记或粗化。

注意，如果你使用的是悬挂节点适应（缺省），那么你将不能建立一个比原来更粗的网格。对于这种情况，你必须使用正则适应。还需要注意的是，正则粗化仅适用于 2D 或轴对称几何问题。详细内容请看第 5.11 节。

6．如果你想设置任何适应选项（在第 5.10 节有介绍），点击 **Controls…**按钮打开 **Grid Adaption Controls** 面板。

7．点击 **Mark**，这样通过把单元放进一个适应记录而为适应（细化/粗化）给单元做标记，或着点击 **Adapt** 立即执行适应。

（如果你希望使细化或粗化不活动，或者为细化或粗化做标记，那么你可以在做标记或适应之前关掉 **Refine** 或 **Coarsen** 选项。）

5.5　各向同性适应

一些流动可能包含一些流动特性，这些特性基于一定量的值是容易区别的。例如，波表示一个总压不足，喷射可以通过相对高速流的一个区域而确定。由于这些区域也包括一些重要流动量（例如湍流中的 k 和 ε）的大梯度，因此在执行一个关于相关流动量的各向同性适应可能比细化个别流动变量的梯度要方便些。

各向同性适应函数允许你标记或细分单元，这些单元位于一个对挑选的场变量函数所规定的范围的内部或外部。网格可以基于几何或解矢量数据被细化或为细化而被标记。特别是，关于场变量的显示列中的任何量都可以被由于各向同性适应。下面介绍了一些关于怎样使用各向同性标记/适应的特性的例子：

使用坐标值或二次曲面函数建立掩饰。

细分单元，其中单元有一个不超出规定范围的速度值。

单元有一个压力或连续残差在一个所要求的范围之外，标记和展示这样的单元以确定数值解快速变化的位置。

用于各向同性适应的函数的途径是为每个单元计算所指定的值（速度，二次曲面函数，质心的 x 坐标，等等），然后访问每个单元，为细化对单元做标记，其中这些单元有在规定范围内（或外）的值。

5.5.1 各向同性适应的例子

各向同性在一个问题中有用的例子如图 5.5.1 所示。关于一个撞击射流的网格与 x 速度等值线一起在图中展示。基于 x 速度的各向同性适应仅允许在射流中的网格细化，细化结果如图 5.5.2 所示。

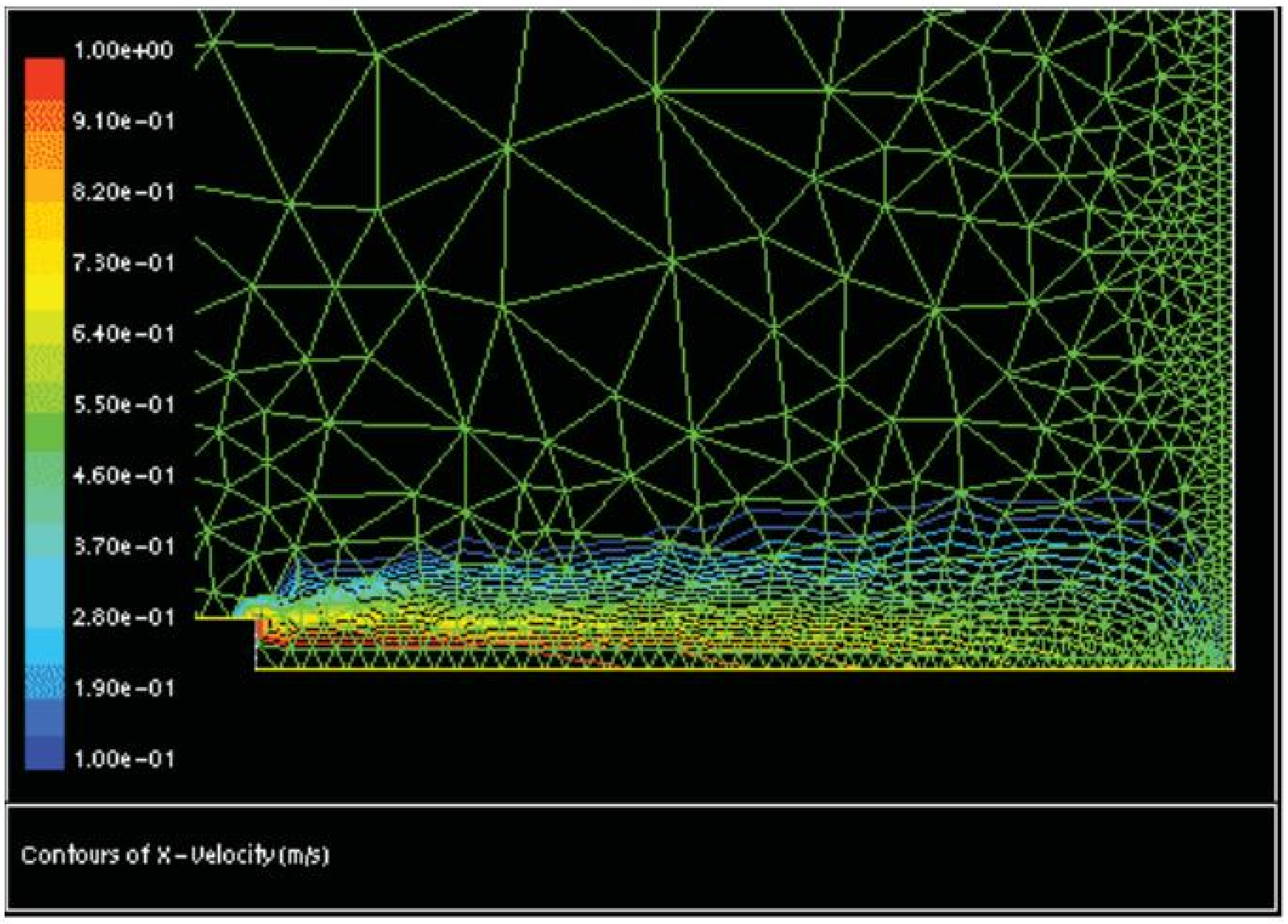


图 5.5.1：在适应前的撞击射流网格和 x 速度等值线

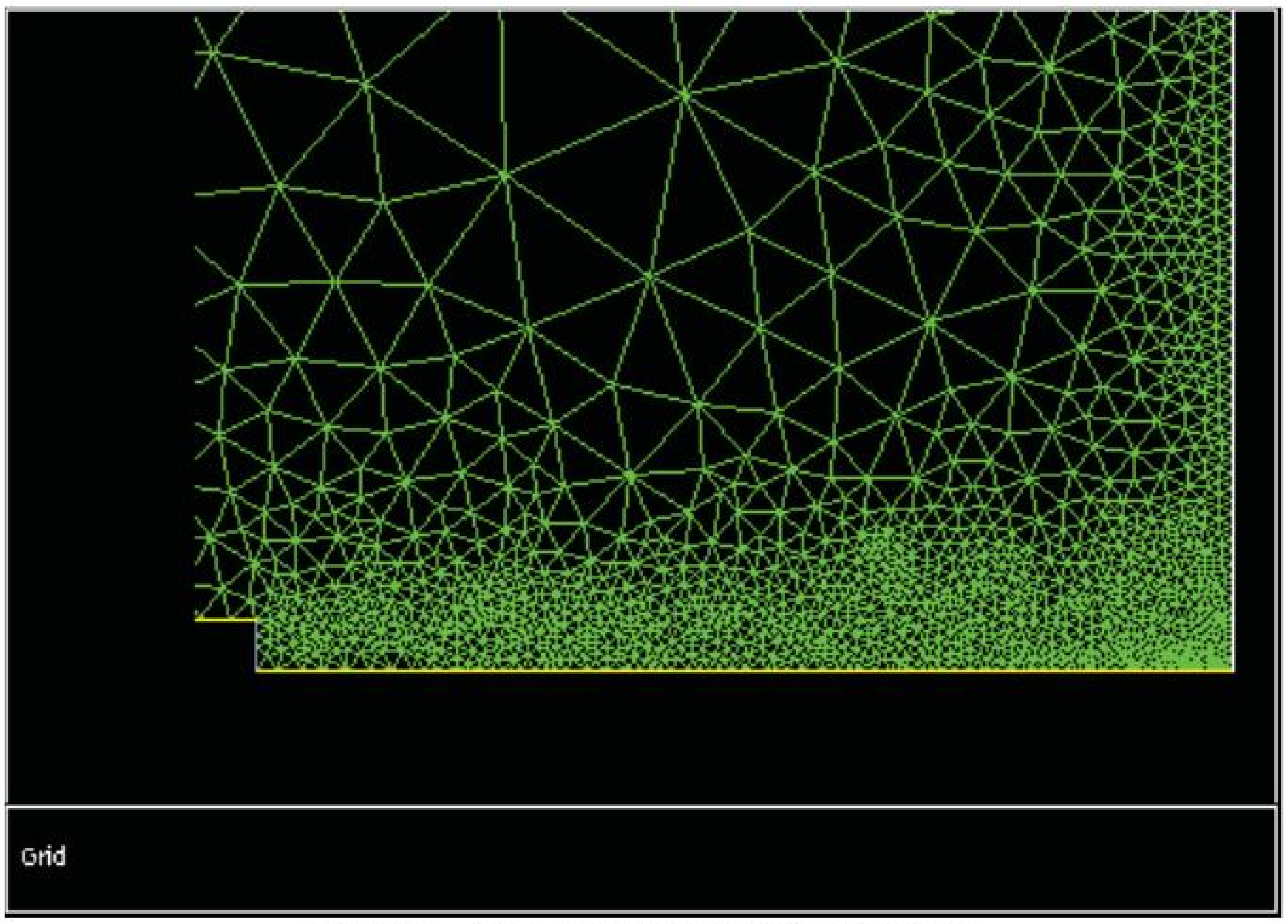
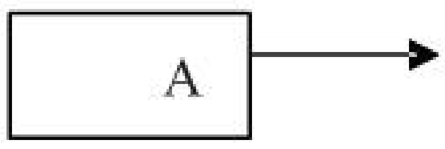


图 5.5.2：各向同性适应之后的撞击射流网格

5.5.2 执行各向同性适应的步骤

在 **Iso-Value Adaption** 面板（如图 5.5.3 所示）中执行各向同性适应。



Iso-Valu

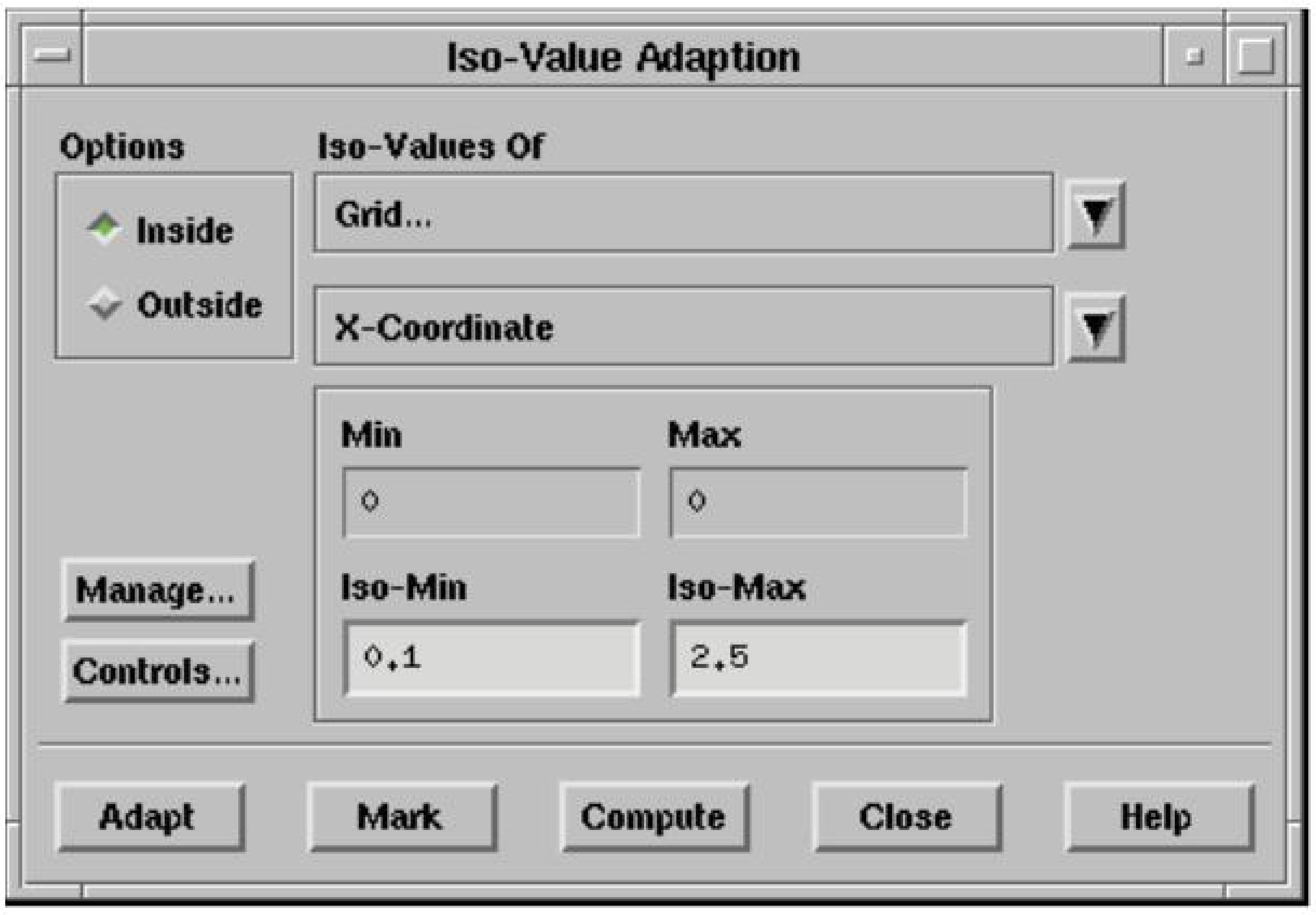


图 5.5.3：Iso-Value　Adaption 面板

执行各向同性适应的一般步骤如下：

1．在 Iso-Value Adaption 面板中，选择 Iso-Values Of 下面的列中所需要的解变量，然后点击 Compute 去更新 Min 和 Max 场。

2．选择 Inside 或 Outside 选项，并设置 Iso-Min 和 Iso-Max 的值。

如果选择 **Inside**，在 **Iso-Min** 和 **Iso-Max** 值之间的各向同性值的单元将被标记或细分。

如果选择 **Outside**，低于 **Iso-Min** 和高于 **Iso-Max** 的各向同性值的单元将被标记或细分。

3．（可选择的）如果你想设置任何适应选项（在第 5.10 节有介绍），点击 **Controls...** 按钮打开 **Grid Adaption Controls** 面板。

4．点击 **Mark**，这样通过把单元放进一个适应记录而为细化给单元做标记，或着点击 **Adapt** 立即执行适应。

5.6　区域适应

如果解域的某个区域需要更细的网格，那么可以使用区域适应来细化网格。区域适应函数标记或细分一个区域内部或外部的单元，其中这个区域是由文本或鼠标输入而定义的。目前，网格可以在一个六面体（2D 中是四边形）、球体（在 2D 中是圆）或圆柱体的内部或外部被细分或标记。基于标记/适应特性的区域对细分区域特别有用。另外，你可以使用区域标记来建立掩饰适应记录，其中掩饰适应记录可以用来限制细化和粗化的范围。

5.6.1 定义一个区域

区域适应函数的基本方法是先定义一个六面体的（四边形的）、球形的（圆的）或圆柱的区域。通过输入定义对角线的两个点的坐标来定义六面体（四边形）。通过输入球心的坐标和半径来定义球（圆）。要定义一个圆柱（cylinder），只要指定定义圆柱轴的点的坐标及半径的长度即可。在 3D 中，这将定义一个圆柱体；在 2D 中，这将是一个任意定位的矩形，其长等于圆柱的轴长，宽等于半径。使用柱面选项定义的矩形与使用四边形选项定义的不同，前者可以在区域中任意的定位，而后者的长和宽必须与坐标轴一致。

你可以在适当的空框键入确切的坐标，也可以使用鼠标选择网格或解场显示的位置。区域被定义之后，每个在指定区域内部/外部有一个质心的单元都被标记为细化。

5.6.2 区域适应的例子

图 5.6.1 中的网格是为求解围绕一个襟翼翼型的流动而建立的。网格在接近翼型表面的地方非常细以便粘性影响的区域可能被解决。我们知道流动分离发生在襟翼的吸气表面，然而，网格在远离翼型时增长的非常快，以至于不能很好的预知流动分离。要解决这个问题，需对襟翼周围的圆形区域（用鼠标指针选取）内的网格进行适应。适应后的网格如图 5.7.2 所示。注意，当区域适应被执行时，关于适应的最小单元体积被限制（在第 5.10 节中有介绍）以阻止表面附近非常小的单元被进一步细分。

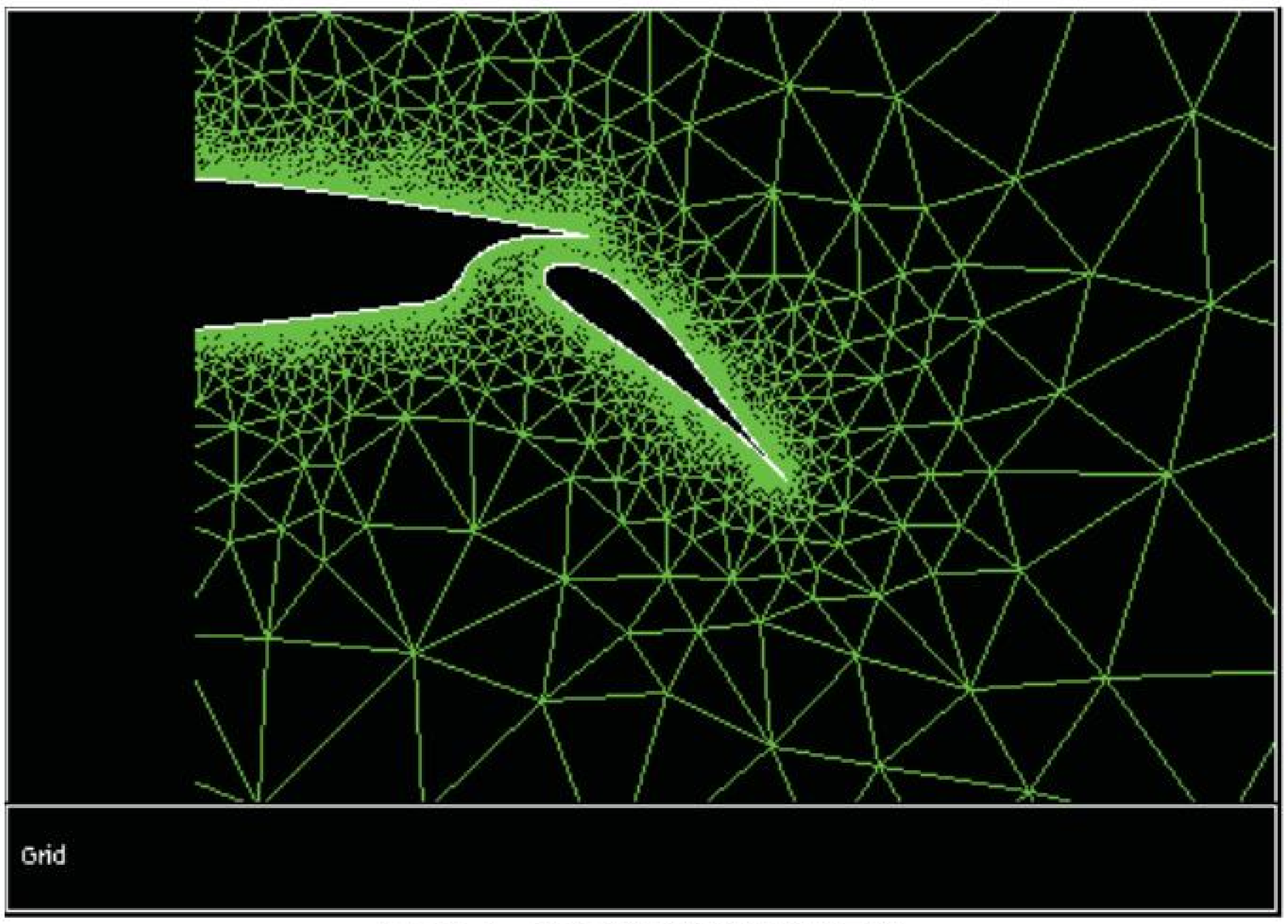


图 5.6.1：适应前的襟翼的翼型网格

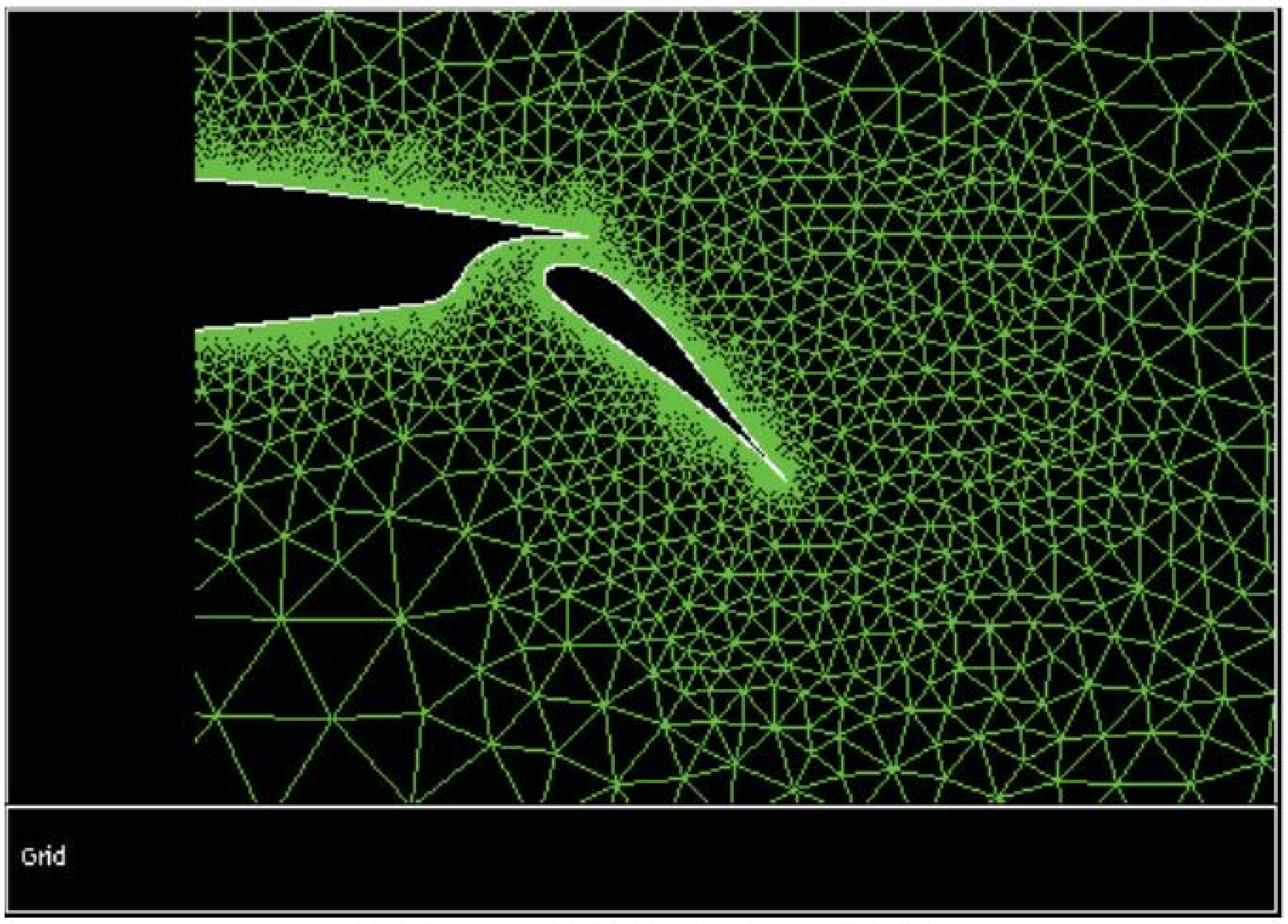
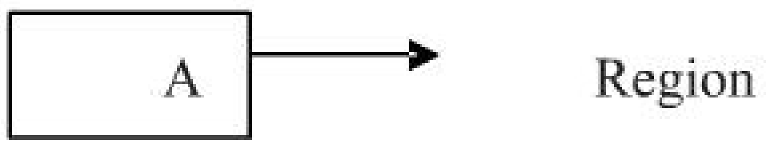


图 5.6.2：区域适应后的襟翼的翼型网格

5.6.3 执行区域适应的步骤

在 Region Adaption 面板中执行区域适应。



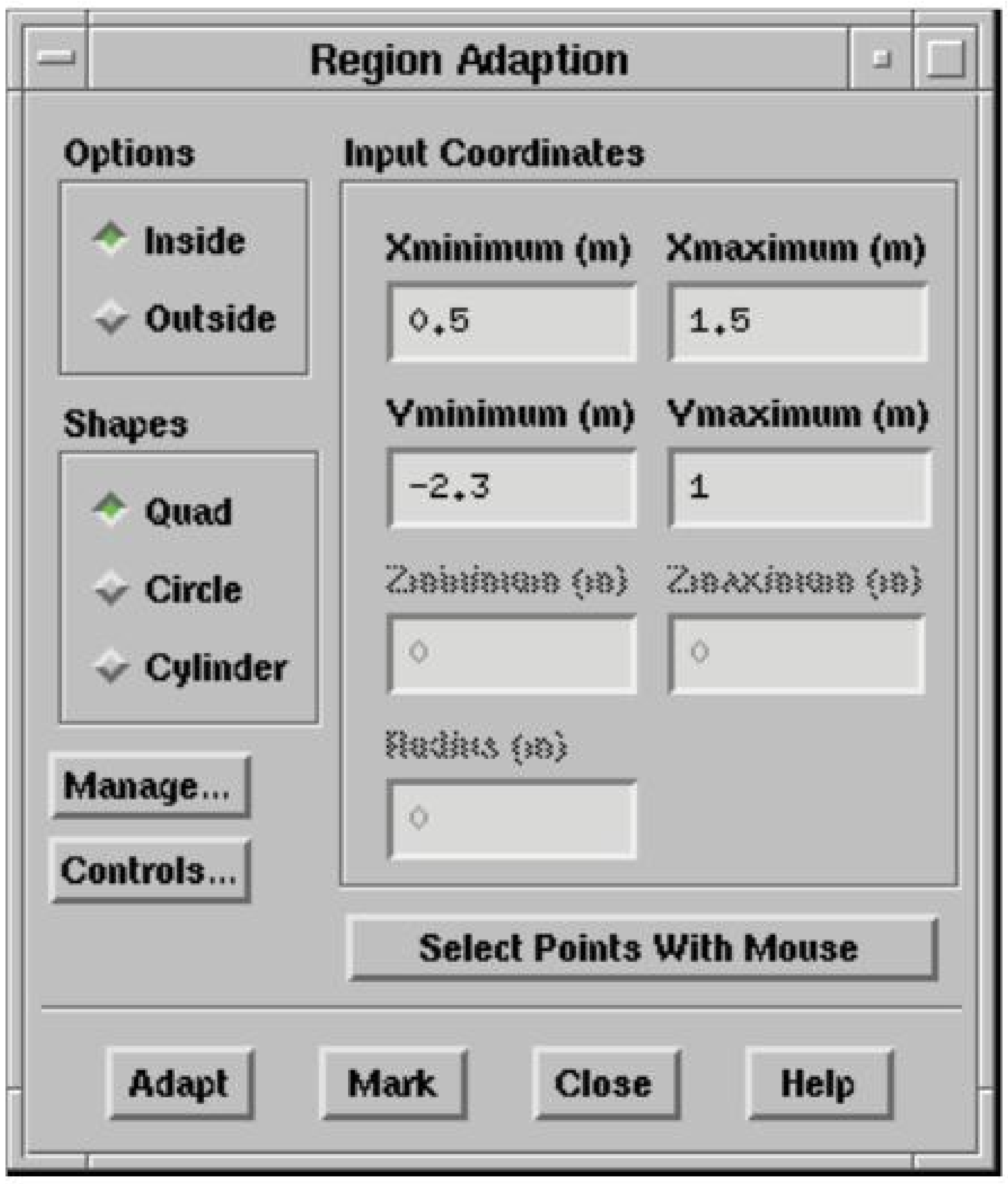


图 5.6.3：Region Adaption 面板

执行区域适应的一般步骤如下：

1．在 Region Adaption 面板中，选择 Inside 或 Outside 选项。

若选择 **Inside**，则在规定的区域内有质心的单元将被标记或细分。

若选择 **Outside**，则在规定的区域外有质心的单元将被标记或细分。

2．规定区域的形状。在 2D 中，可以选择一个 Quadrilateral，Circle，或 Cylinder。在 3D 中，可以选择一个 Hexahedron，Sphere，或 Cylinder。

3．通过把值键入面板或使用鼠标来定义区域。在面板中如下输入：

要定义一个六面体或四边形，则需要输入定义盒子对角线的两个点的坐标：对一个六面体，（Xminimum, Yminimum, Zminimum）和（Xmaximum, Ymaximum），或对一个四边形，（Xminimum, Yminimum）和（Xmaximum, Ymaximum）。

要定义一个球或圆，则需要球的球心坐标（Xcenter, Ycenter, Zcenter）或圆心坐标（Xcenter, Ycenter），及半径。

要定义一个圆柱，则在 3D 中，需要输入定义圆柱轴的坐标的最小值和最大值－（X-Axis Min, Y-Axis Min, Z-Axis Min）和（X-Axis Max, Y-Axis Max, Z-Axis Max），或在 2D 中，输入（X-Axis Min, Y-Axis Min）和（X-Axis Max, Y-Axis Max），不论 2D 还是 3D，都要输入圆柱的 Radius。（在 2D 中，圆柱的轴长和半径分别是矩形的长和宽。）

要使用鼠标定义区域，则点击 Select Points With Mouse 按钮。使用鼠标探针（缺省时是鼠标左按钮），你可以从一个网格和解场的展示中选择输入坐标。在你选择了点之后，值将被自动地下载到面板的适当的框中。在标记或适应之前，你可以对你想修改的值进行编

辑。

要定义一个六面体或四边形，你可以按任意顺序选择对角线的两个点。

要定义一个球或圆，首先选择质心的位置，然后选择一个位于这个球或圆上的点（即，这点表示远离轴的半径）。

要定义一个圆柱，首先选择定义圆柱轴的两个点，然后选择一个点，它是远离轴的半径。

4.（可选择的）如果你想设置任何适应选项（在第 5.10 节有介绍），点击 **Controls...** 按钮打开 **Grid Adaption Controls** 面板。

5. 点击 **Mark**，这样通过把单元放进一个适应记录而为细化给单元做标记，或着点击 **Adapt** 立即执行细化。

5.7 体积适应

如第 5.1 节所述，一个网格中的单元体积渐进的变化对于解的精确性及收敛性是非常有益的。如果这种网格的建立或适应过程导致一个网格内没有这种特性，那么可以在单元体积或单元和它相邻的单元之间的体积变化的基础上，通过使用选项为细化的体积适应来改进网格。

5.7.1 方法

基于体积的大小标记或细分网格被经常用来切除大单元或全局地细化单元。其过程是标记体积大于规定的临界值的任何单元为细化。

基于单元体积内的变化的标记或细分被用来改进网格的光滑性。其过程是标记体积大于规定的临界值的任何单元为细化。体积变化是通过轮循过面和比较单元邻域与面的比率而计算的。例如，在图 5.7.1 中，比率 $V1/V2$ 和比率 $V2/V1$ 与临界值做比较。如果 $V2/V1$ 大于临界值，那么 C 被标记为细化。

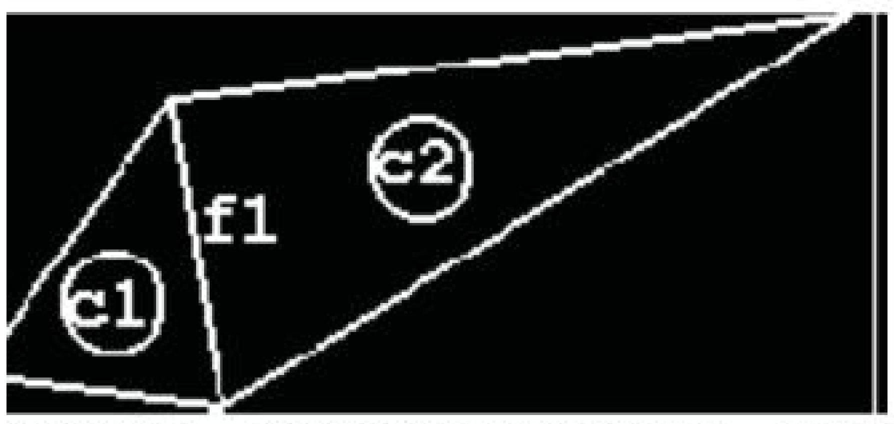


图 5.7.1：作为邻近一个面的单元体积的比率，体积变化被计算

5.7.2 体积适应的例子

图 5.7.1 是为计算一个湍流喷射而建立的网格。在 TGrid 中使用局部细化建立了一个网格，在射流区域中网格是比较细的，而在其它区域则比较粗糙。在射流边上的单元体积有一个急剧的变化。

要改进网格，使用体积适应来细化网格，其中体积适应有一个关于单元体积的变化最不应超过 50% 的标准。对适应的单元体积变化的最小值也有限制。体积适应之后的网格，经过光滑和交换，所得的网格如图 5.7.3 所示。从图中可以看到，射流内的细化区域与周围的网格之间的界面不再那么明显。

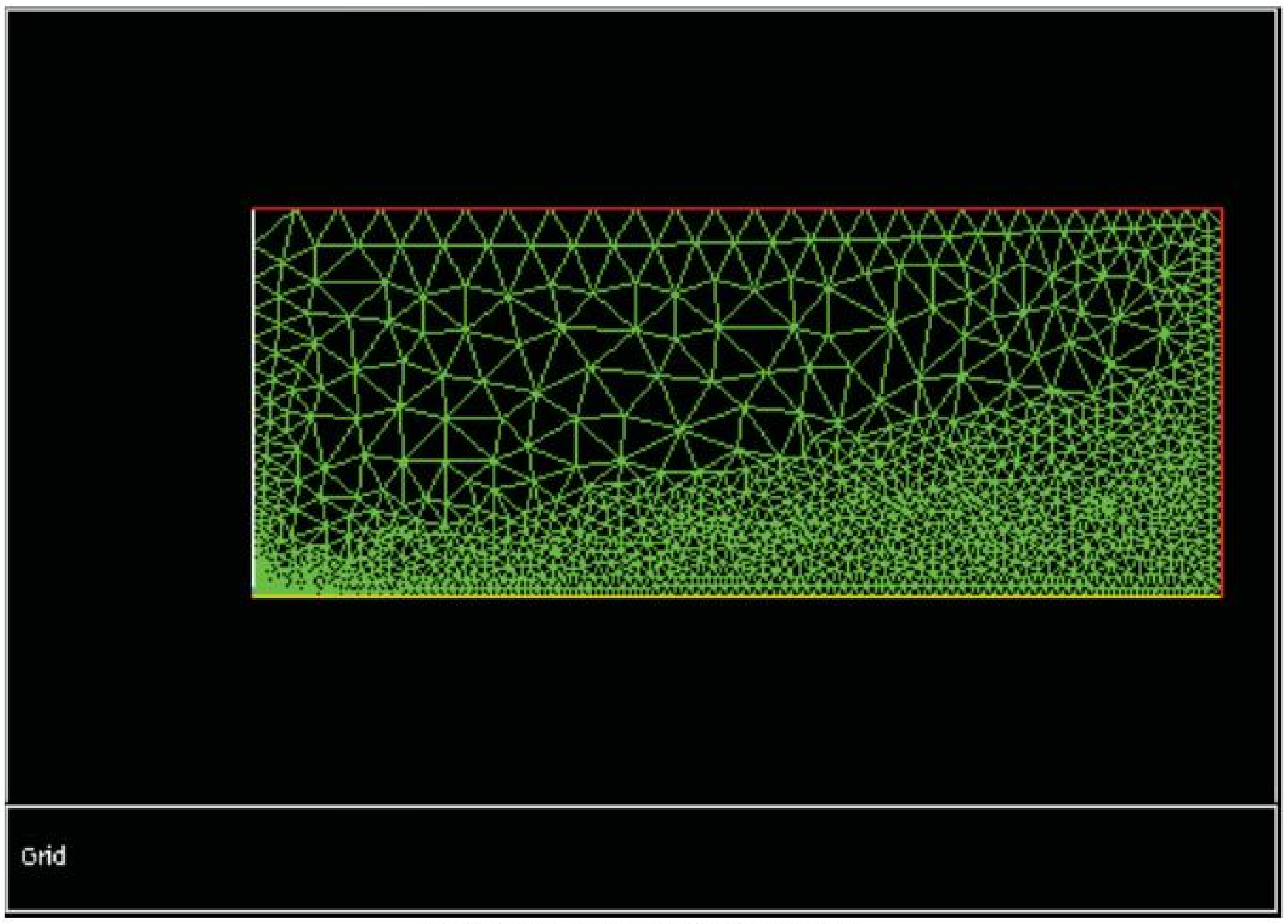


图 5.7.2：适应前的射流网格

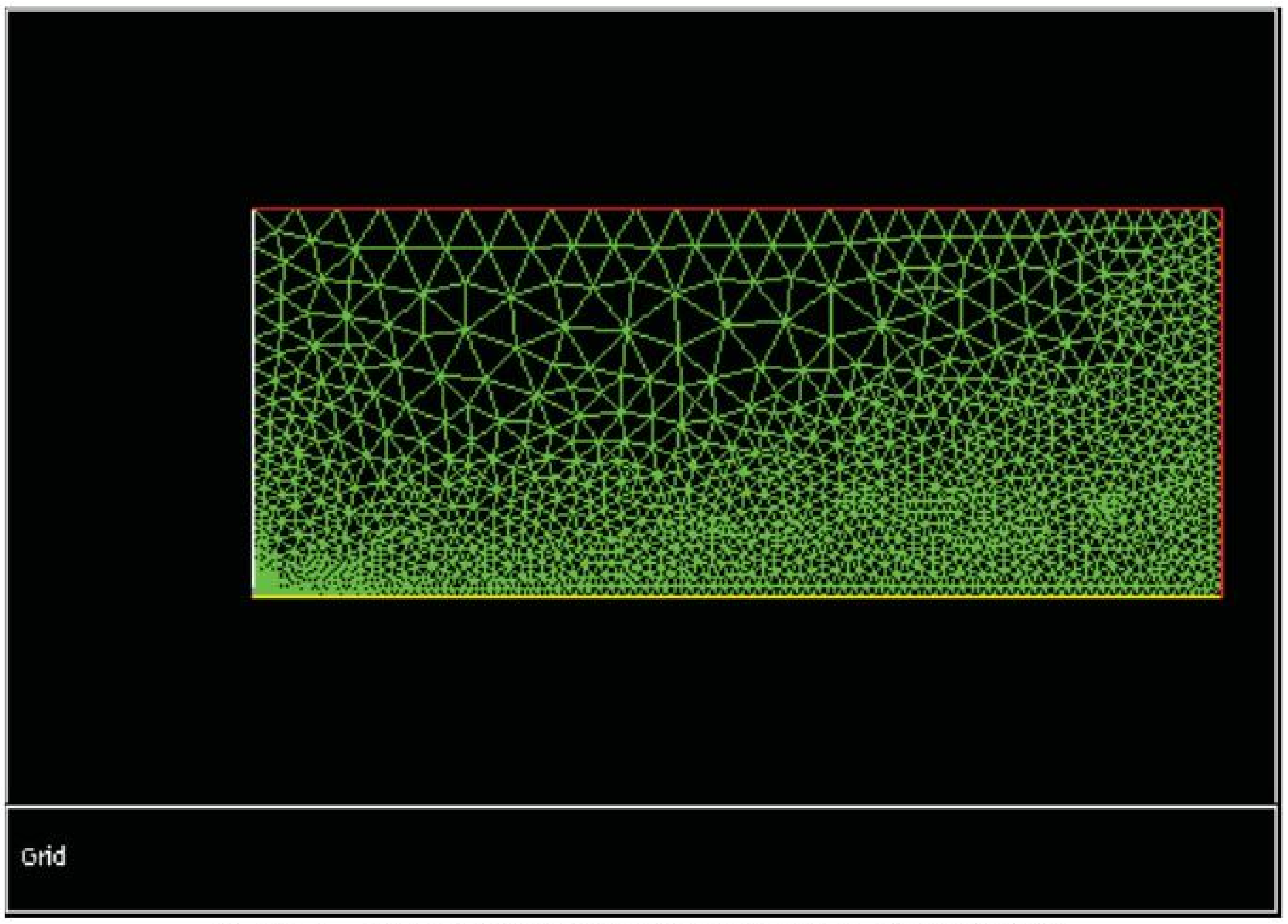


图 5.7.3：基于单元体积的变化的体积适应之后的射流网格

5.7.3　执行体积适应的步骤

在 Volume Adaption 面板（如图 5.7.4 所示）中执行体积适应。

A → Volume

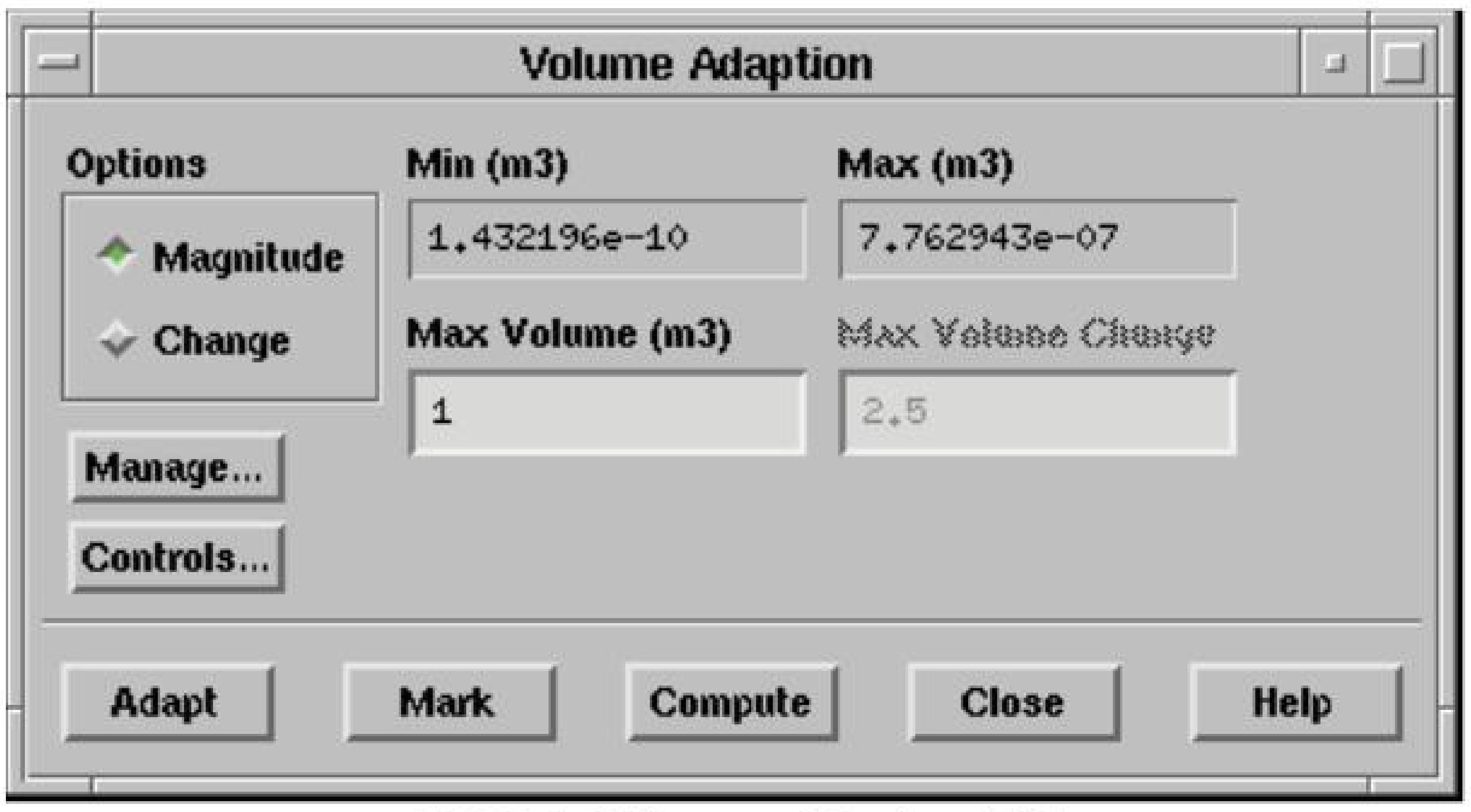


图 5.7.4：Volume　Adaption 面板

执行体积适应的一般步骤如下：

1．在 **Volume Adaption** 面板中，通过选择 **Magnitude** 或 **Change** 选项而规定在体积大小的基础上还是在体积变化的基础上进行适应。

2．点击 Compute 更新 Min 和 Max 框中的值。它们将显示单元体积或单元体积变化的范围（即最小值与最大值），只有哪种的范围当然要取决于第一步的选择。

3．设置 Max Volume 或 Max Volume Change 的值。如果基于体积 Magnitude 进行适应，那么体积大于 Max Volume 的单元将被标记或细化。如果基于体积变化进行适应，那么体积变化大于 Max Volume Change 的单元将被标记或细分。

4．（可选择的）如果你想设置任何适应选项（在第 5.10 节有介绍），点击 **Controls...** 按钮打开 **Grid Adaption Controls** 面板。

5．点击 **Mark**，这样通过把单元放进一个适应记录而为细化给单元做标记，或着点击 **Adapt** 立即执行细化。

5.8 y+和 y*适应

FLUENT 为湍流近壁模型（标准的壁面函数，非平衡的壁面函数，和增强的壁面处理）提供了三种选项。

当建立网格时，由于测量近壁分解的需求往往比较困难，y+和 y*适应能够使你在解的过程中沿壁面适当地细化或粗化网格。如果你正在使用增强的壁面处理，你将对 y+进行适应；如果你正在使用壁面函数，你既可以对 y+也可以对 y*进行适应。

5.8.1 方法

方法是计算关于在规定的粘性壁面区域上的边界单元的 y+或 y*，定义 y+或 y*所允许的最小值和最大值，及标记或适应合适的单元。y+或 y*的值低于所允许的最小临界值的单元将被标记为粗化，y+或 y*的值高于所允许的最大临界值的单元将被标记为细化（除非粗化或细化没有被激活）。

5.8.2 y+适应的例子

图 5.8.1 给出了一个管道流动的网格，其中顶部边界是壁面，底部边界是一个对称平面。在一个初始解之后，确定在壁面边界上的单元的 y+值太大，那么 y+适应被用来细分单元。适应后的网格如图 5.8.2 所示。从图中可以看出，在细化过程中单元沿壁面边界的高已被降低了。然而，细化后的单元尺寸分布没有原来网格的整齐，它是 y+适应所产生不利效果。

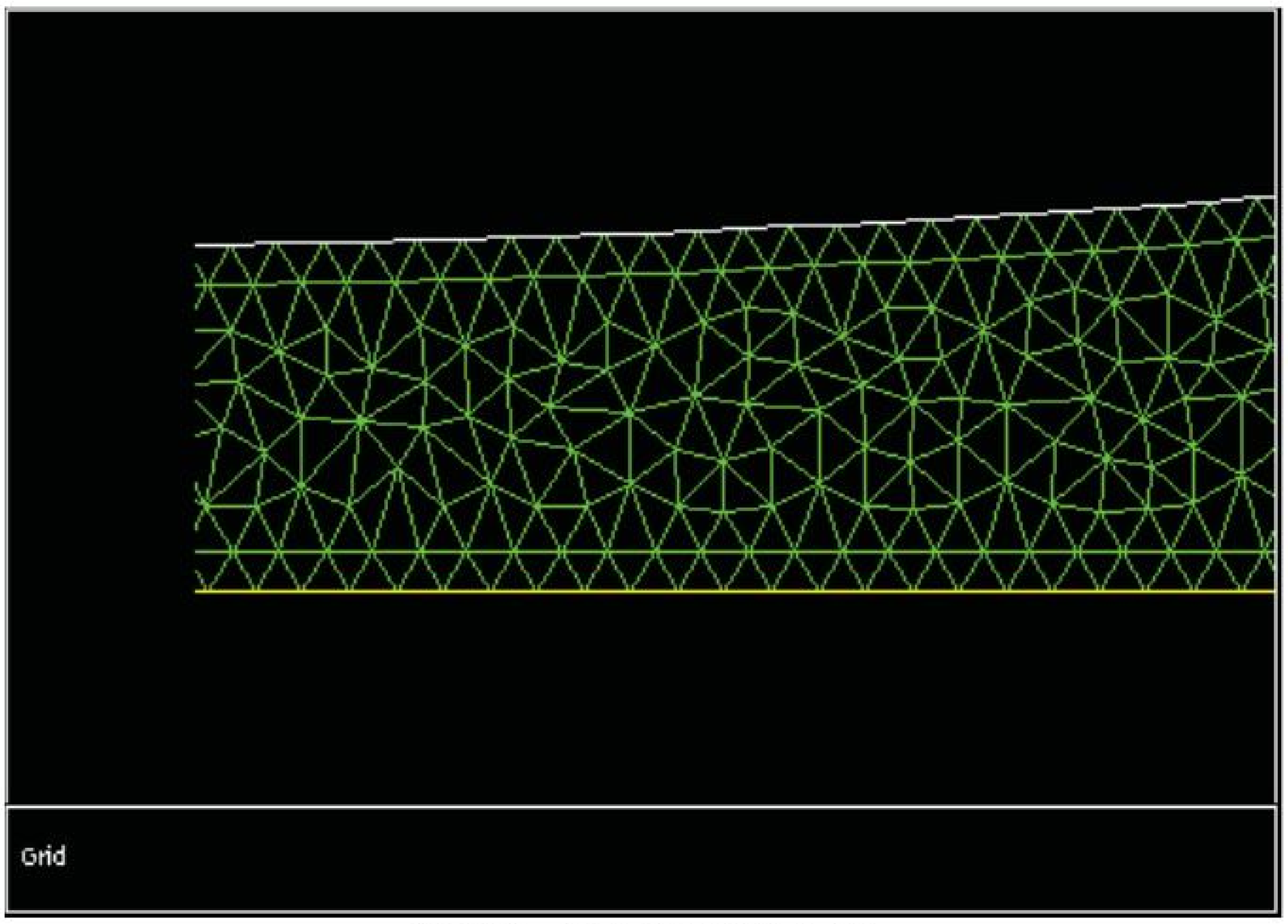


图 5.8.1：适应前的管道流网格

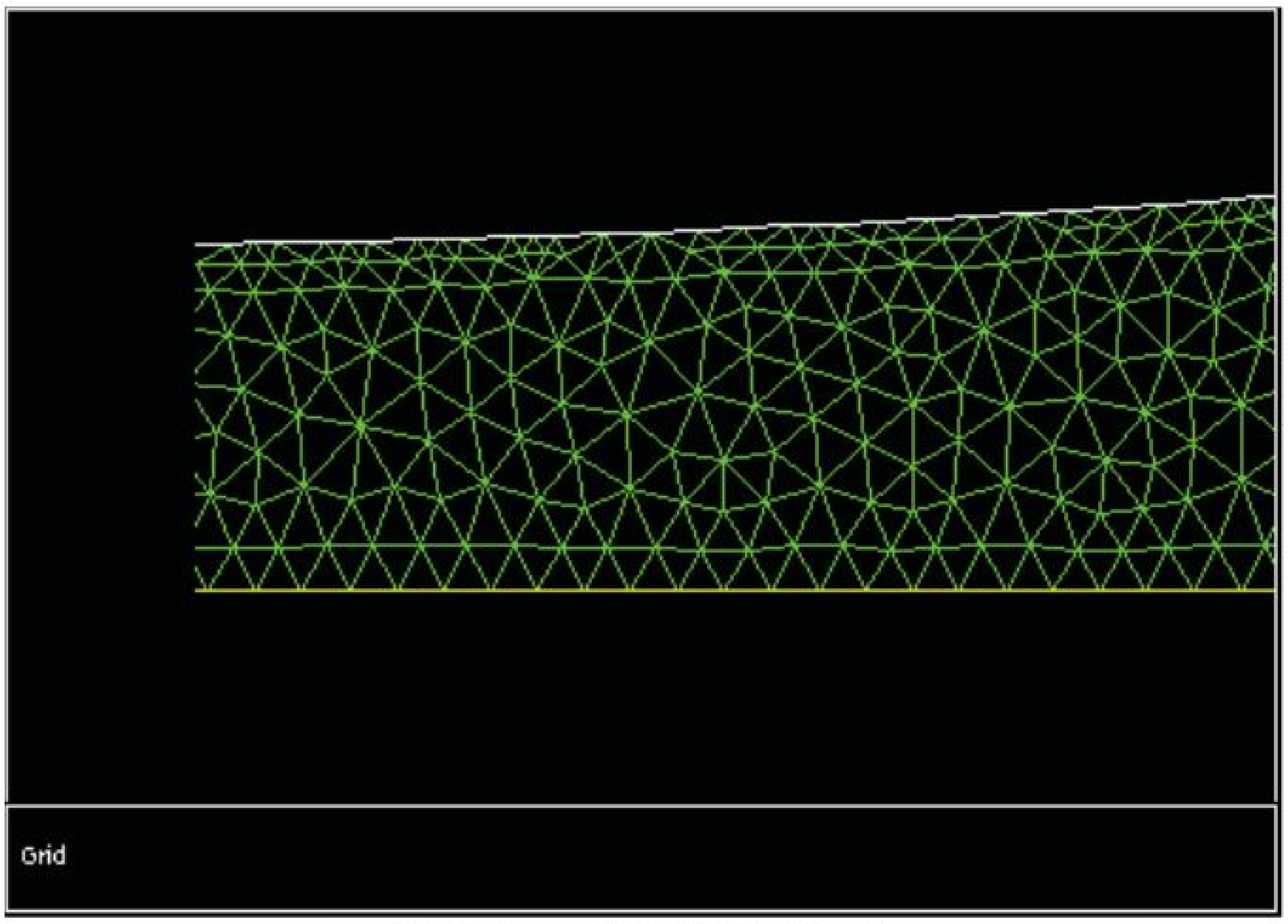
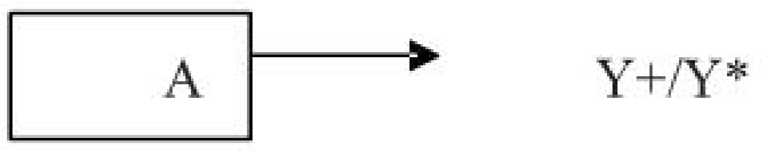


图 5.8.2：y+适应后的管道流网格

5.8.3 执行 y+或 y*适应的步骤

在 **Y+/Y*Adaption** 面板中执行 y+或 y*适应（如图 5.8.3 所示）。



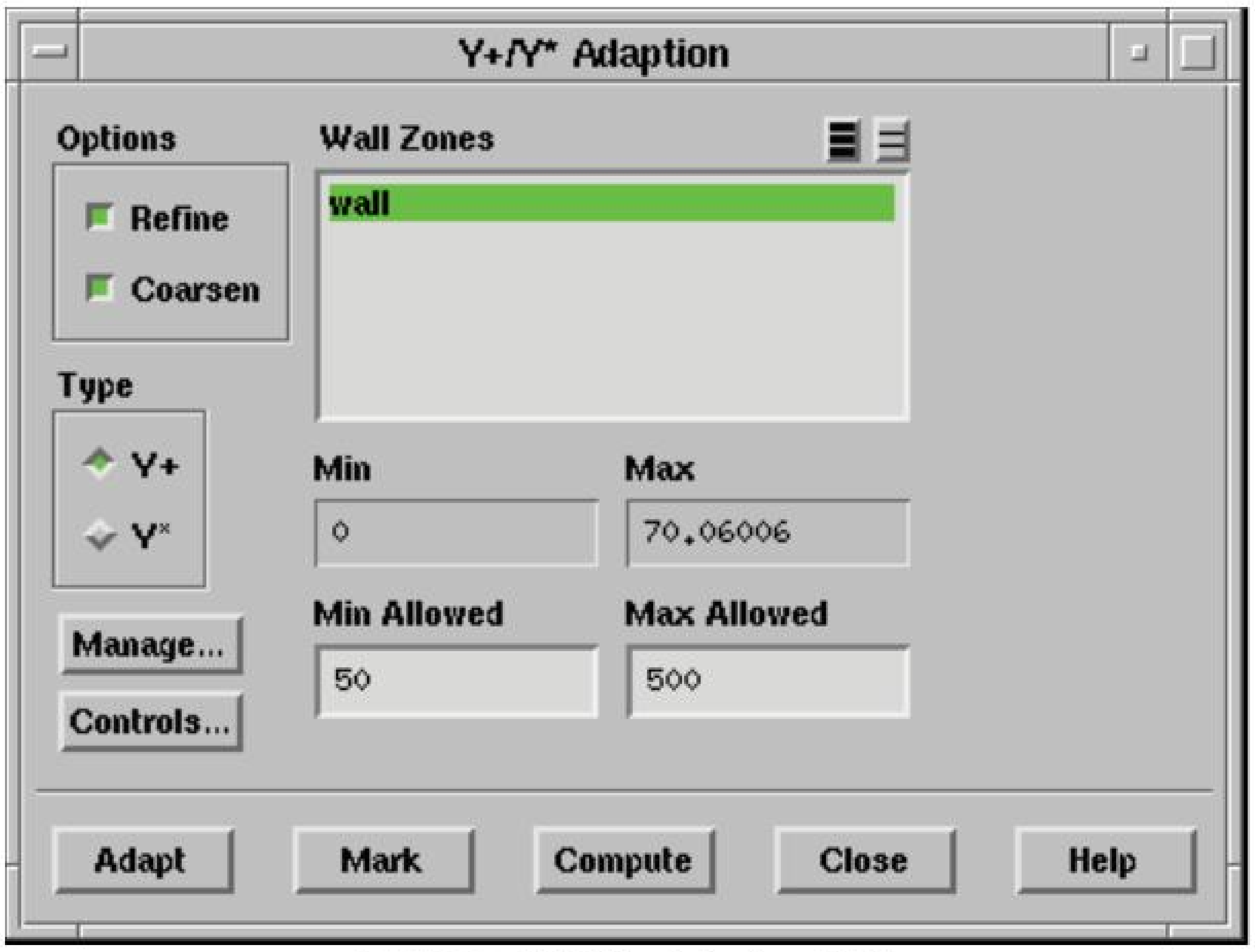


图 5.8.3：Y+/Y*　Adaption 面板

执行 y+或 y*适应的一般步骤如下：

1．在 **Y+/Y* Adaption** 面板中，选择适应的类型 **Y+**或 **Y***。（若使用增强的壁面处理，则选择 **Y+**；若使用壁面函数，则两种类型皆可选择。）

2．在 **Wall Zones** 列中选择壁面区域，其中壁面区域上的边界单元是你相对其进行标记或适应的。点击 **Compute** 更新 **Min** 和 **Max** 框中的值。注意，所显示的值是关于所有壁面区域的最小值和最大值，而不只是那些所选壁面区域的。

3．设置 **Min Allowed** 或 **Max Allowed** 的值。y+或 y*的值小于 Min Allowed 的值的单元将被粗化或标记为粗化；y+或 y*的值大于 Max Allowed 的值的单元将被细化或标记为细化。

注意，如果你正在使用悬挂节点适应（缺省时），那么你将不能建立一个比原来的网格更粗的网格。这时，你必须使用正则适应。还需要注意的是，正则粗化仅适用于 2D 或轴对称几何。相关内容在第 5.10 节有介绍。

4．（可选择的）如果你想设置任何适应选项（在第 5.10 节有介绍），点击 **Controls…** 按钮打开 **Grid Adaption Controls** 面板。

5．点击 **Mark**，这样通过把单元放进一个适应记录而标记单元为适应（细化/粗化），或着点击 **Adapt** 立即执行适应。

5.9　管理适应记录

你可以操作、删除和展示通过标记单元为适应而建立的适应记录。由于这些记录将被用来调整网格，因此操作它们的功能为你提供了对适应过程的额外控制。

在 Manage Adaption Registers 面板中执行适应记录的管理（如图 5.9.1）。

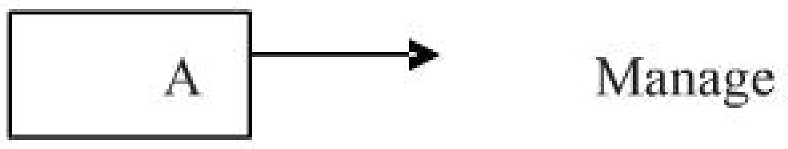




图 5.9.1：Manage Adaption Registers 面板

（你也可以在任何适应面板中通过点击 **Manage…**按钮打开这个面板。）

一般来说，想把适应局限在一个特定的区域（适应标记）和/或建立一个更准确的误差指示器是激发建立混合适应函数的因素。FLUENT 提供了几个帮助建立混合适应函数的工具。第一，你可以使用几何的和/或解的矢量信息来建立初始的适应记录。在建立了适应记录之后，你可以操作这些记录和与它们相关的细化和粗化标记。改变记录的类型和/或把记录结合起来以建立所需要的混合函数，从而达到操作记录的目的。标记是通过使用

Exchange，Invert，Limit，和 Fill 来操作的。最后，你可以删除，展示，和最重要的一点即适合混合适应函数。

5.9.1 操作适应记录

用于修改和操作适应记录的基本的工具有三个：改变类型，合并和删除。

改变记录的类型

目前，有两种类型的记录：适应记录和掩饰记录。目前的适应函数通过 GUI 被访问，它和文本界面产生适应记录－即，指定被标记为细化或粗化的单元的记录。这些记录可以被转换成掩饰记录，不过，要通过改变它们的类型实现。不像适应记录，掩饰记录仅保持两种状态：ACTIVE 和 INACTIVE。如果适应记录被转换成一个掩饰的，那么被标记为细化的单元使 ACTIVE，其它的单元都是 INACTIVE；即不考虑被标记为粗化的单元。一般地，被转换成掩饰的适应记录是由专门把单元标记为细化的适应函数所产生的那些记录，例如区域或各向同性适应函数。适应和掩饰记录的结合的方式也是它们的主要的不同之处。

把一个或多个记录从适应改变成掩饰类型（反之亦然）的步骤如下：

1．在 **Registers** 列中选择记录。

2．点击 Register Actions 下面的 Chang Type。

新类型的记录作为 **Type** 将出现在 **Register Info** 下面的框里。你可以单独地选择每个记录来查看它当前的类型。

合并记录

在每个适应记录被建立和适当地修改之后，这些记录被合并起来以建立混合适应函数。用下面的方式可以合并任意数目的记录：

把所有的适应记录合并成一个新的适应记录。

把所有的掩饰记录合并成一个新的掩饰记录。

把新适应记录和新掩饰记录合并起来。

用下面的方式可以合并任意数目的适应记录：

如果单元在任意的记录中被标记为细化，那么在新记录中标记单元为细化。

如果单元在任意的记录中被标记为粗化，那么在新记录中标记单元为粗化。

合并掩饰记录的方式类似于细化标记：如果任何单元被标记为 ACTIVE，那么在新的记录中的单元被标记为 ACTIVE。

最后，在一个适应和掩饰记录的合并中，仅有在掩饰记录中被标记的单元可以在合并的记录中有一个适应标记。

例如，基于压力梯度建立一个适应函数可能在整个解域中产生标记为细化和粗化的单元。如果那时这个记录与一个掩饰记录合并，其中这个掩饰记录是从一个球内部的被标记的单元建立的，那么仅有在球内部的单元将在新记录中被标记为细化或粗化。

掩饰的影响依赖于它们被应用的顺序。

合并两个或多个记录的步骤是：

1．在 **Registers** 列中选择记录。

2．点击 Register Actions 下面的 Combine 按钮。

所选的记录将保持完好无缺，来自合并的记录将被添加到 **Registers** 列中。在某些例子中，三个新的记录可能被建立：适应记录的一个合并，掩饰记录的一个合并，和两个合并的记录的一个合并。

删除记录

适应记录被删除的数目是任意的。删除记录的主要原因是放弃不需要的适应记录，以减少可能的混乱，同时，删除这些无价值的记录可以降低生成不需要结果的潜能。另外，只有 32 个适应记录可以同时存在。所以，你可能需要删除不需要的记录以为新的腾出空间。

永久的删除一个或多个记录的步骤是：

1．在 **Registers** 列中选择记录。

2．点击 Register Actions 下面的 Delete 按钮。

5.9.2 修改适应标记

适应标记是用来标记一个单元应当被细分、粗化还是中立（既不细分也不粗化）的标识符。目前，修改适应标记的基本工具有四个：**Exchange**，**Invert**，**Limit**，和 **Fill** 操作。

Exchange 操作把所有标记为细化的单元改变成标记为粗化的单元，把所有标记为粗化的单元改变成标记为细化的单元。一般地，这个操作适应于只有细化标记的适应记录。例如，交换操作可以被用来粗化一个矩形区域。首先，你建立一个把一矩形区域的单元标记为细化的适应记录。然后，使用 **Exchange** 操作修改单元标记，从而建立一个矩形区域，且其中的单元被标记为粗化。

Invert 操作仅可以用于掩饰记录。这个操作使得所有 ACTIVE 标记的单元转化到 INACTIVE，所有 INACTIVE 标记的单元转化到 ACTIVE。例如，如果你生成了一个定义一圆形区域的掩饰，你可以使用 **Invert** 操作快速地修改这个掩饰以定义圆外部地区域。

Limit 操作对所选择的适应记录应用现在的适应体积限制。一般地，你使用这个操作确定现在的限制对适应过程的影响。你可以通过仅对细化大单元设置限制来使用体积限制以建立一个更均匀的网格。在所有的单元达到一相同尺寸之后，为获得要想的分解，你可以继续细化过程。

最后，**Fill** 操作把在适应记录中没有被标记为细化的单元标记为粗化的。如果你正则把多个记录合并成一个新的记录，那么也许你想使用 **Fill** 操作。当你合并记录的时候，一个单元当且仅当在所有的记录中被标记为粗化的时候才将被标记为粗化。如果你建立一个适应记录，且它有一个仅把单元标记为细化的操作，但你不想禁止粗化，那么你应当在把这个记录与任何其它的记录合并之前使用 **Fill** 操作。

修改适应标记的步骤是：

1．在 **Registers** 列中选择记录。

2．点击 Mark Actions 下面的 Exchange，Invert，Limit 或 Fill 按钮。

5.9.3　展示记录

查看单元的标记在建立混合适应的过程中往往是很有用的。你可以策划一个在单元的质心的标记和/或一个单元的接线框以查看单元的状态。缺省时，标记为细化的单元被涂以红色，而粗化的则被涂以蓝绿色。另外，在一个掩饰记录中，标记为 ACTIVE 的单元也被涂以红色。这些是标记为适应的单元，但添加到网格中或从网格中减去的单元，其最后的个数将取决于适应的限制和网格的特性。

展示一个记录的步骤是：

1．在 **Registers** 列中选择记录。

2．若需要，点击 **Options…**，设置任何展示选项（下面有介绍）。

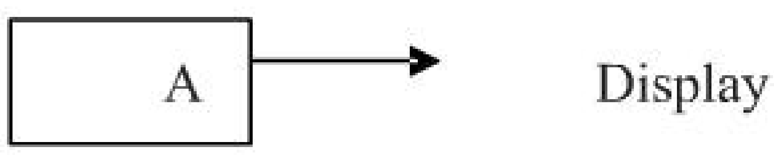
3．点击 **Display** 按钮。

适应的展示选项

适应记录展示的各个方面都可以被修改，例如接线框的可视性，留下阴影、标记的可视性，颜色，大小，符号，以及所展示的表面或面网格。

适应记录展示的能力允许你查看被标记为适应的单元。你可以使用标记和/或接线框展示在一个适应或掩饰记录里的被做标记的单元。标记是被放在单元质心的一个符合，包括细分标记和粗化标记。你可以改变这些标记的符合、颜色和大小、一个连接框由三角形或四面体的边组成，它的颜色和各自的标记颜色相同，并且它还可以用此颜色填充。

最后，网格的部分可以用标记符号或接线框画出，这有助于估测标记的单元的位置。



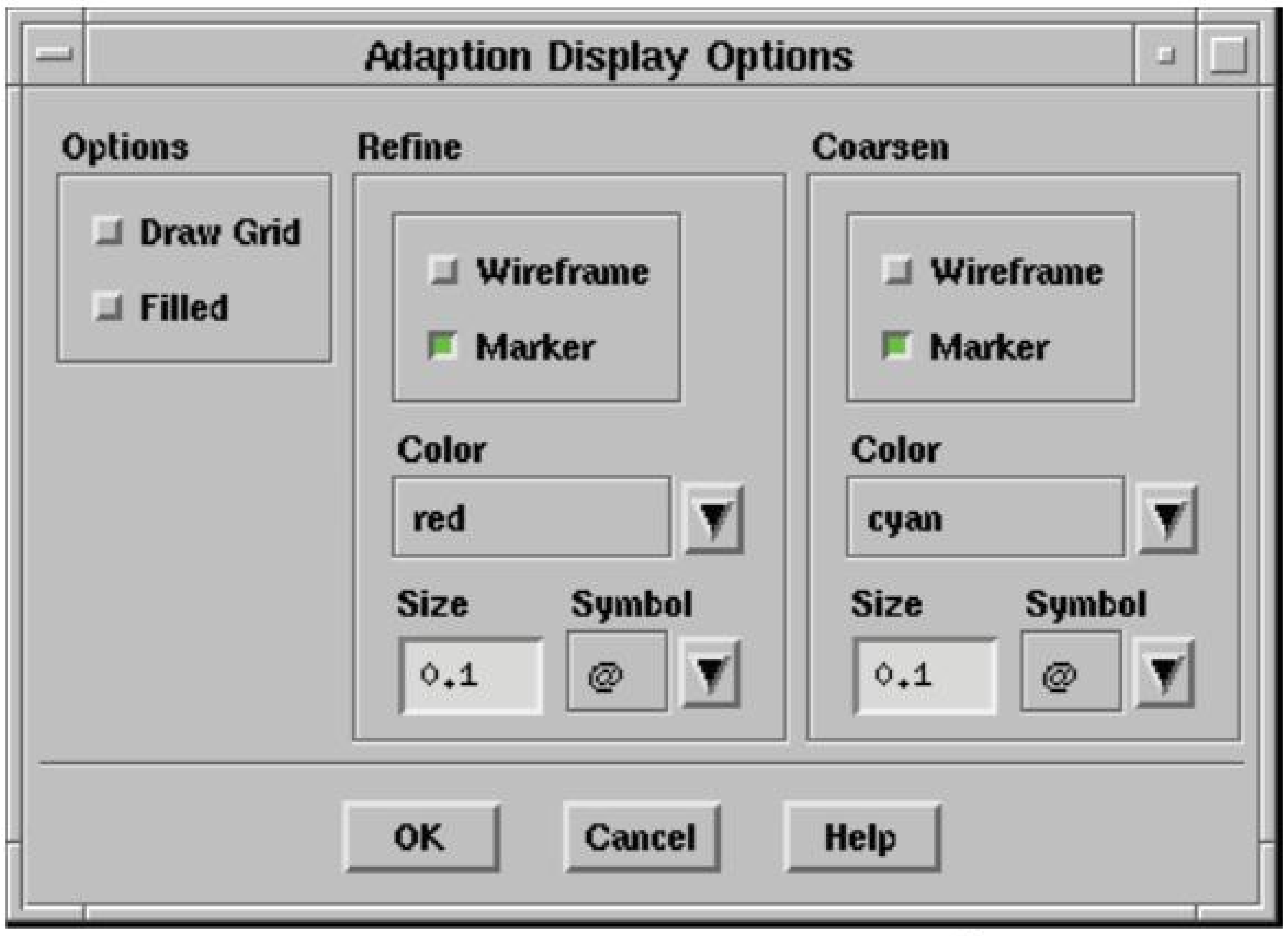


图 5.9.2：Adaption Display Options 面板

（你也可以通过点击 Manage Adaption Registers 面板中的 Options…按钮打开这个面板。）

要激活或不激活关于被标记为细化/粗化的单元的接线框的展示，请打开或关掉 **Refine** 和/或 **Coarsen** 下面的 **Wireframe** 选项。若要填充接线框，则打开 **Filled** 选项即可。

要激活或不激活关于被标记为细化/粗化的单元的标记的展示，请打开或关掉 **Refine** 和/或 **Coarsen** 下面的 **Marker** 选项。如果你使用标记，那么你可以在 **Size** 的框里规定它们的大小，在 **Symbol** 下面的列中规定它们的符合。

要改变细化或粗化的标记/接线框的颜色，则在 **Refine** 或 **Coarsen** 下面的 **Color** 下面的列中选择一个新颜色。缺省时，细化标记/接线框是红色，粗化标记/接线框是蓝绿色。

要展示在记录展示中的网格的一部分，则激活 **Draw Grid** 选项。**Grid Display** 面板将自动地出现，你可以在这里设置网格展示的参数。当你在 **Manage Adaption Registers** 面板中点击 **Display**，在 **Grid Display** 面板中定义的网格展示将被包含在记录展示中。

5.9.4 适应记录

调整网格的主要目的是有效地提高解的精确度。这些记录工具为你提供了建立混合的适应函数能力。最后，用户化的适应函数被用来管理细化和粗化的网格。

执行适应的步骤如下：

1. 选择 **Registers** 列中的记录。
2. 点击 **Adapt** 按钮。

5.10 适应性控制

FLUENT 允许改变适应的类型—即从悬挂节点到正则，反之亦然，或者对单元区域、可以被调整的单元的尺寸、以及从适应过程中产生的单元数目给予约束。你也可以修改梯度函数中的体积权重的强度，限制细化和/或粗化的适应过程，及在正则粗化过程中，控制哪些节点适合被从网格中删除。

控制适应的这些方面的所有参数是在面板 **Grid Adaption Controls**（如图 5.10.1 所示）中设置的。

A → Display

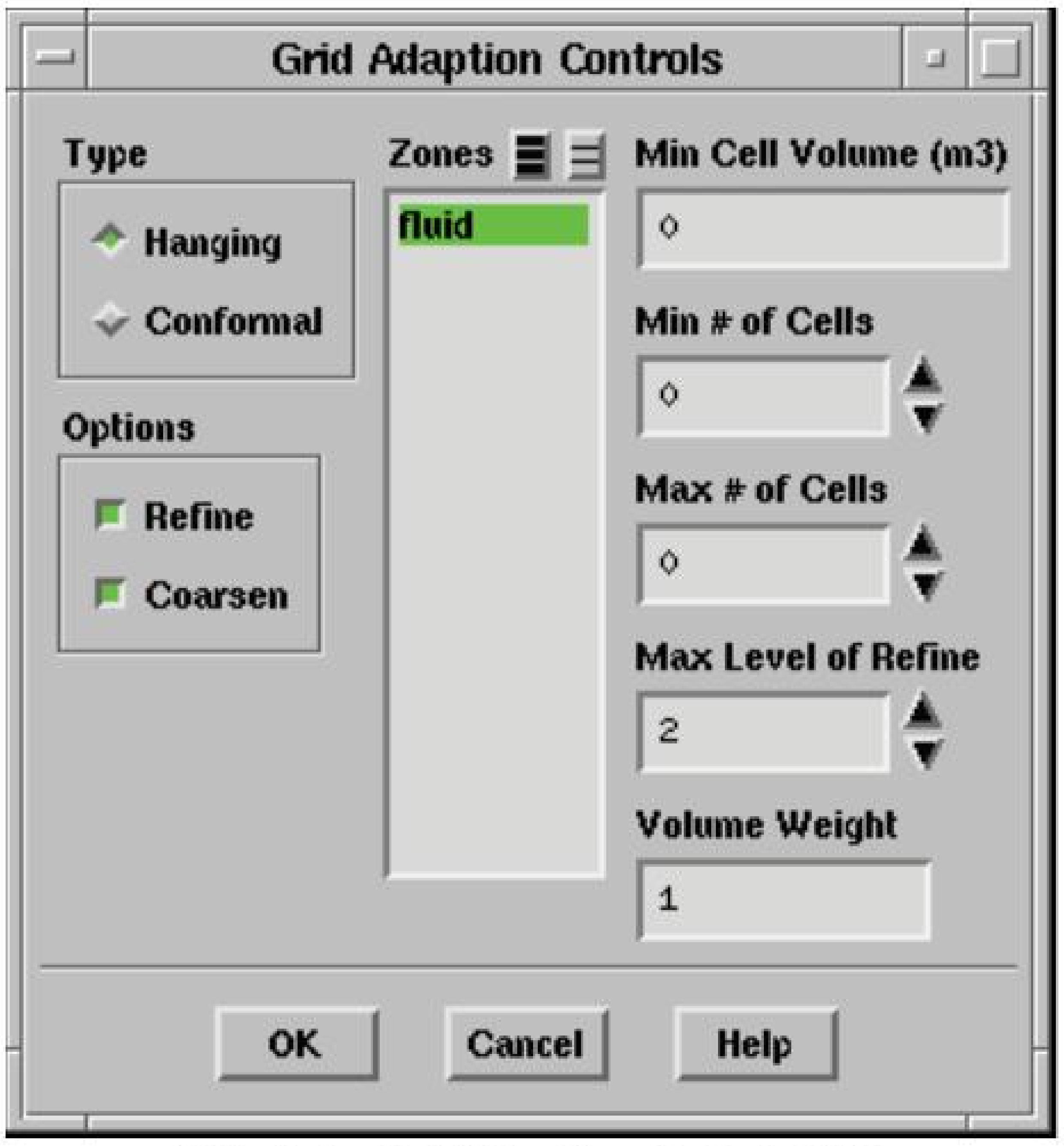


图 5.10.1：Grid Adaption Controls 面板

（也可以在任何一个适应面板中通过点击 **Controls…**而打开这个面板。）

控制适应的类型

你可以选择使用悬挂节点适应还是正则适应。你也可以限制适应的过程，其中包括经过细化增加网格的分解，经过粗化消减网格密度。

要使用悬挂节点适应（缺省），则选择 Type 框中的 Hanging，要使用正则适应，则选

择 Conformal。关于悬挂节点和正则的适应程序在第 5.2 节中有具体介绍。

要激活/不激活细化，则打开/关闭 **Refine** 选项。

要激活/不激活粗化，则打开/关闭 **Coarsen** 选项。

借助域限制适应

你可以把适应过程限制到规定的单元域。一般地，构成流体和固体区域的单元有非常不同 分解要求和误差指示器。通过把适应限制到一个特定的单元域，你可以使用不同的适应函数以建立最佳网格。

要把适应限制到一个特殊的单元域，则在 **Zones** 列中选择单元域（你想在这些单元域中执行适应）。缺省时，适应将在所有单元域中被执行。

借助单元体积或体积重量限制适应

最小单元体积的界限把细化过程限制在体积大于界限的单元中。你可以使用它在较大的单元上开始细化过程，逐渐地降低界限以建立一个均匀单元尺寸的分布在 **Min Cell Volume** 的框里设置这个界限。注意，对于一个 2D 轴对称问题，在这个框中输入的则是最小单元面积。

另外，梯度体积重量可以被修改。一个零值排除了体积权重的情况，1 使用整个体积，介于 0 到 1 的值则依比例决定体积权重。在 **Volume Weight** 的框中设置这个值。

限制单元的总数

你可以在 **Max # of Cells** 的框中设置网格中所允许的最大单元总数。在 **Min # of Cells** 的框中设置网格中的最小单元个数。缺省时的零值对单元数目没有限制。

在悬挂节点适应过程中控制细化层面

在非正则适应过程中，细化被用来切开单元。通过设置 **Max Level of Refine**，你可以控制此细化层面的个数。对于大多数问题，缺省值 2 是一个好的出发点。如果这个值不够，那么你可以增加这个值，但是请记住，即使在一个 2D 问题中，在适应的区域，缺省值 2 也能够以 16 的因数增加单元的个数。一个 0 值使得这个参数极大：在这种情况下，你应当对 **Min Cell Volume** 使用一个适当的限制。请看第 5.2.2 节关于悬挂节点适应的更详细的相关内容。

在正则粗化过程中控制节点的切除

通过修改节点切除的记号，你可以控制粗化过程中的节点切除。节点切除记号仅适用于正则适应。对于悬挂节点适应，仅细化节点能够在粗化过程中被切除，及它们一直被删除。

为保证区域边界的原来形状被保持，通常仅由细化引进的节点被切除。例如，考虑一

个矩形区域的网格。如果在矩形边界上的一个节点被切除，区域的形状不改变，但是如果矩形的一个角上的节点被删除，形状则从一个四边形变成一个五边形。

通过改变节点切除记号，你可以修改缺省行为。大部分普通的修改允许原始的内部节点的切除。除去内部节点不破坏边界形状。事实上，如果原始网格在一个区域中有实质的分解，而在物理特性上没有或有很小的变化，那么这是非常有益的。

缺省时，仅允许细化节点的切除，正如下面所指示的：在 Grid Adaption Controls 面板中，激活 Boundary-Refined，Internal-Refined 和 Periodic-Refined，不激活 Boundary-Original Internal-Original 和 Periodic-Original。如果你想使这些类型的节点的切除处于不活动状态，那么你可以关掉相关的控制按钮。类似地，如果你想激活它们的切除，打开相关的控制按钮。

5.11　用光滑和交换的方式改善网格

通常情况下，光滑和面交换会提高最后数值网格的质量，所以说它们是补助网格适应的工具。光滑重新配置节点和面的交换修改单元的连通性，从而在质量上取得改善。

！面交换仅对三角形和四面体单元的网格适应。

使用 **Smooth/Swap Grid** 面板（如图 5.11.1 所示）来操作光滑和交换。

A → Smooth/Swap…

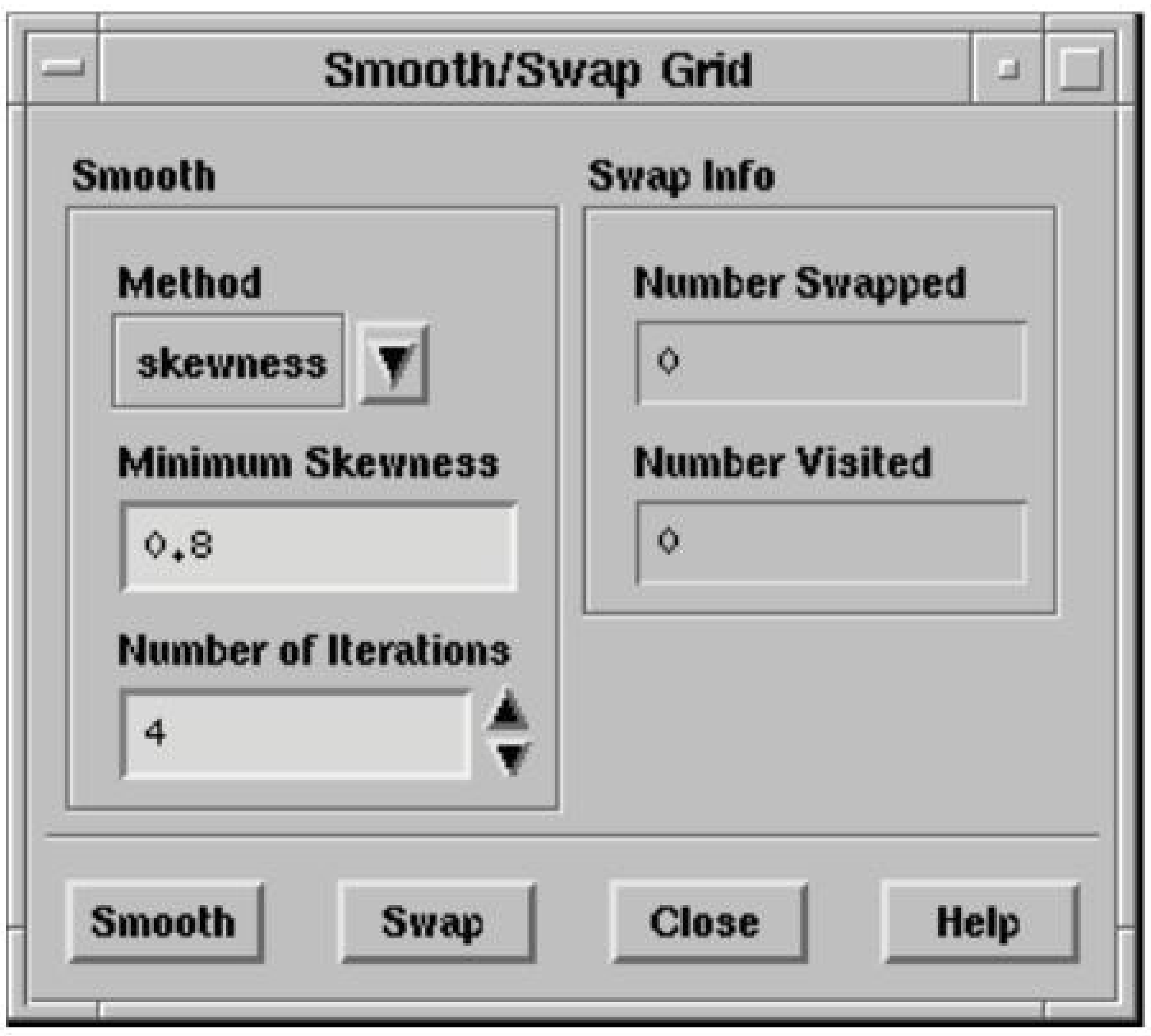


图 5.11.1：Smooth/Swap Grid 面板

5.11.1 光滑

在 FLUENT 中关于光滑的方法有两种：Laplace 光滑和基于偏斜的光滑。第一种方法可以用于各种类型的网格，而第二种方法仅适用于三角形/四面体网格。

！建议对三角形和四面体网格一直使用基于偏斜的光滑方法；而把 Laplacian 方法仅使用于四边形和六面体网格。

Laplacian 光滑

当你使用 Laplace 方法时，一个 Laplacian 光滑操作被应用于非结构网格以重新配置节点。新节点的位置是它周围节点位置的平均。松弛因子（介于 0.0 到 1.0 的一个数）乘上计算所得的节点位置的增量。松弛因子取 0 值时节点没有移动，取单位 1 时则节点的移动等于节点位置的整个增量。对于四边形单元的一个典型的结构，图 5.11.2 给出了新节点的位置。

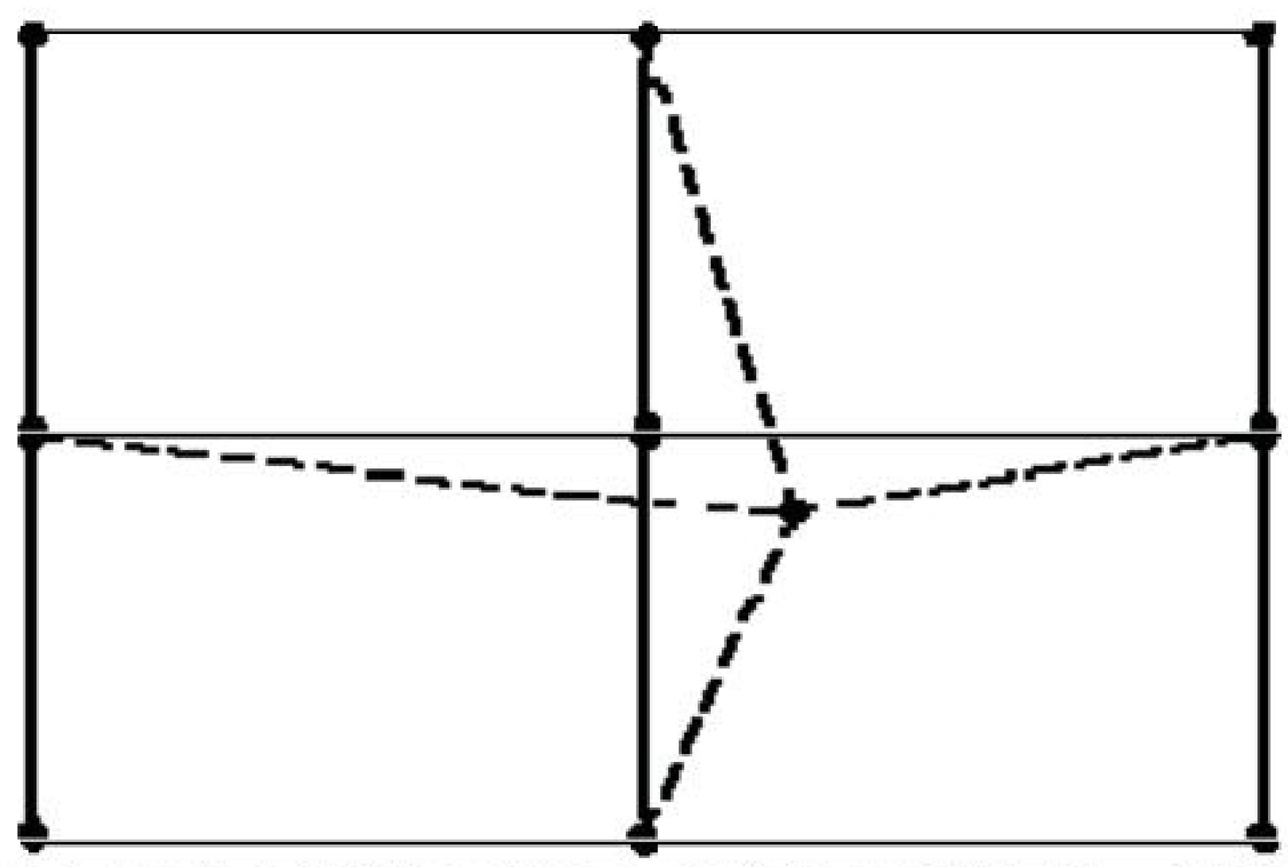


图 5.11.2：关于节点位置的光滑操作的结果图（虚线表示原来的网格，实线表示最后的网格）

重配置策略倾向改善网格的歪斜，但通常对节点的聚类放松约束。在极端的情况下，目前的操作可能建立越过边界的网格线，从而建立了负单元体积。特别是你使用一个大的松弛因子执行了多个光滑操作，那么这最有可能发生在接近急剧拐角或粗糙解决的凸起的拐角的地方。图 5.11.3 是未松弛的光滑迭代之前的一个初始四面体网格，通过它建立互相超出的网格线（如图 5.11.4 所示）。

缺省的光滑参数被安排来以最小的反面效果改进网格质量，但在光滑网格之前，你应当记得保存一个case文件。如果在一个二维问题中，你应用了一个保守的松弛因子，并且

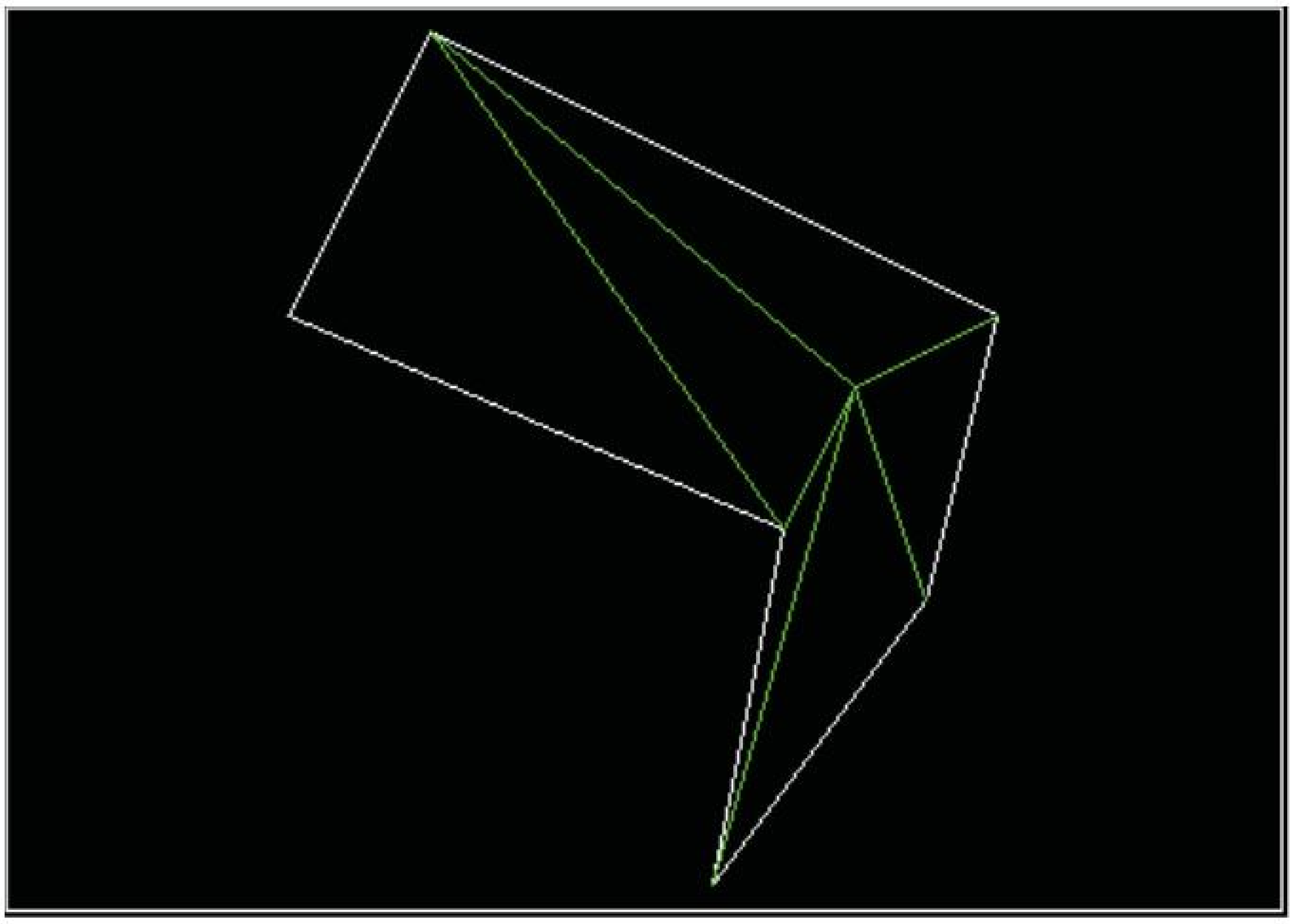


图 5.11.3：光滑操作前的原始网格

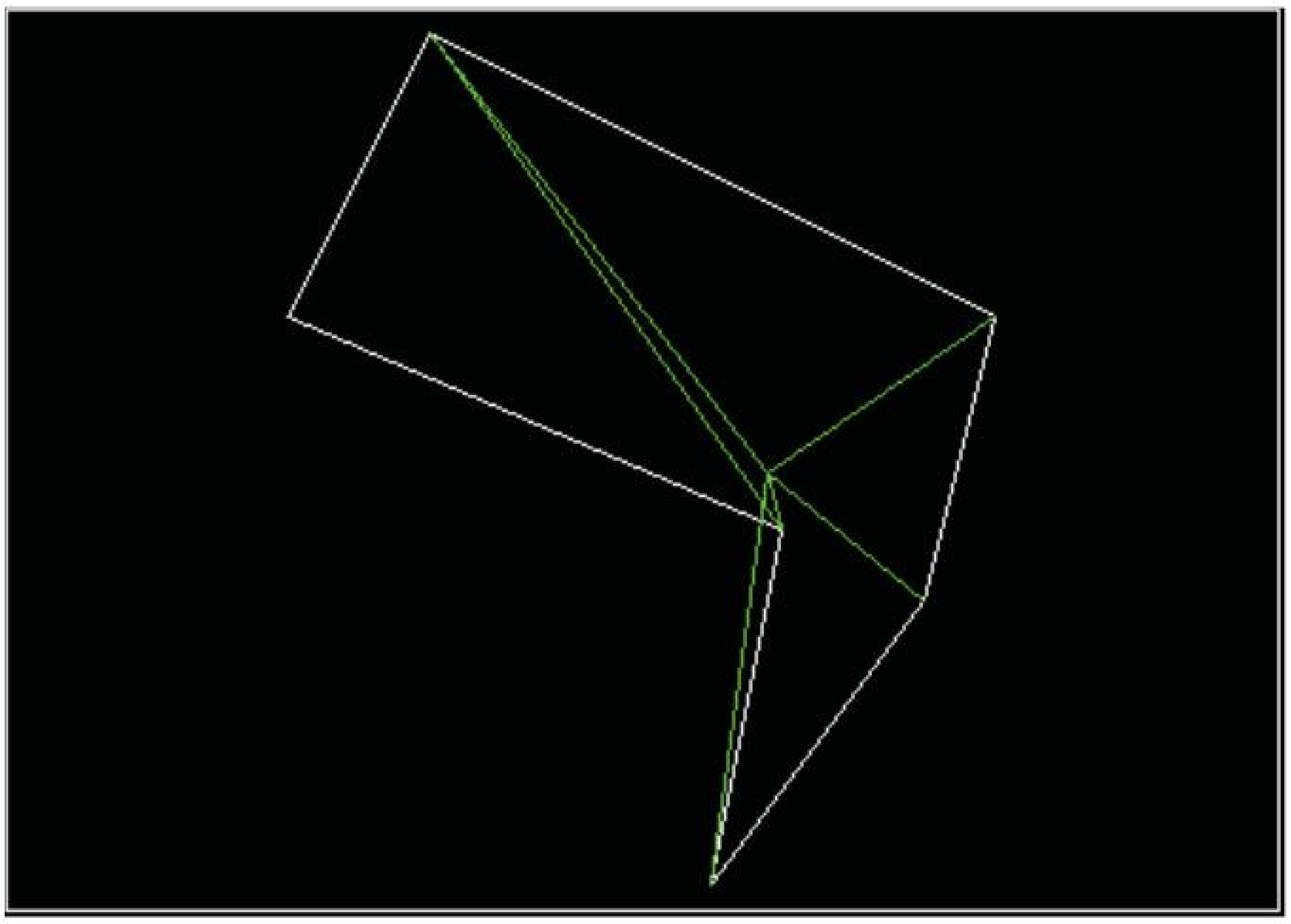


图 5.11.3：网格光滑会导致网格线交叉

缺省的光滑参数被安排来以最小的反面效果改进网格质量，但在光滑网格之前，你应当记得保存一个 case 文件。如果在一个二维问题中，你应用了一个保守的松弛因子，并且

从一个高质量的初始网格出发，那么以后的失败是由光滑极低造成的。然而，网格拓扑的恶化在三维问题中发生的要频繁的多，特别是四面体网格。对于三角形和四面体网格建议用基于歪斜的光滑操作。

光滑操作也可以被重复的应用，但是随着光滑扫过次数的增加，节点有被从边界拉走的趋势，网格倾向于丢失任何聚集的特性。

Laplacian 光滑的步骤

执行 Laplacian 光滑的步骤如下：

1．在 Smooth/Swap Grid 面板中（如图 5.11.1 所示），选择 Smooth 下面的 Method 下面列中的 laplace。

2．在 **Relaxation Factor** 框中，设置因子。因子越低，在节点运动中降的越多。

3．在 **Number of Iterations** 框中，规定光滑相继扫过的网格个数。缺省值为 4。

4．点击 **Smooth** 按钮。

基于歪斜的光滑

当你使用基于歪斜的光滑时，FLUENT 对网格应用一个光滑操作，重新配置内部节点以降低网格的最大歪斜。FLUENT 将努力移动内部节点以改善这些大于规定的“最小歪斜”的单元的歪斜。这个过程非常耗费时间，所以你应当只在高歪斜的单元上执行光滑。通过几次执行光滑节点，可以获得改善的结果。如果移动一个节点导致最大歪斜增大，那么内部控制将阻止它被移动，但是当一个高歪斜的节点被移动时，某些单元的歪斜将被增加是普遍现象。所以，你可能看到在最大歪斜减小的同时平均歪斜增大。**!** 你应当仔细考虑这样做是否对改善网格有意义。仅对带有很高歪斜（例如，0.8 或 0.9）的单元执行光滑也许会缓解对平均歪斜所产生的反面效果。

! 基于歪斜的光滑仅适用于三角形和四面体网格。

基于歪斜的光滑的步骤

执行基于歪斜的光滑的步骤如下：

1．在 Smooth/Swap Grid 面板中（如图 5.11.1 所示），选择 Smooth 下面的 Method 下面列中的 skewness。

2．在 **Minimum Skewness** 框中，设置最小的单元歪斜值。FLUENT 将努力移动内部节点以改进歪斜值大于这个值的单元的歪斜情况。缺省时，对于 2D 情况，**Minimum Skewness** 被设置为 0.4，对于 3D 则为 0.8。

3．在 **Number of Iterations** 框中，规定光滑相继扫过的网格个数。缺省值为 4。

4．点击 **Smooth** 按钮。

基于歪斜的光滑应当与面的交换（看第 5.11.3 节）交替。

5.11.2 面的交换

面的交换可以被用来改善一个三角形或四面体网格的质量。

三角形网格

关于三角形网格的方法是使用 Delaunay 圆周测试来判定是否一个面应当被交换，其中两个三角形单元共享这个面。如果一个单元的外接圆不包含第二个单元中的未被共享的节点，那么这对共享一个面的单元满足圆周测试。图 5.11.5 给出了关于满足和不满足圆周测试的两个单元邻居的例子。在圆周测试不满足的情况，对角线或面被交换，如图 5.11.6 所示。

重复使用面交换技术将产生一个强迫的 Delaunay 网格。如果一个网格是 Delaunay 网格，那么它是唯一的一个最大化网格中的最小角的分成三角形。所以，这个分成三角形趋向于等边单元，为给定的节点分布提供了最好的等边网格。

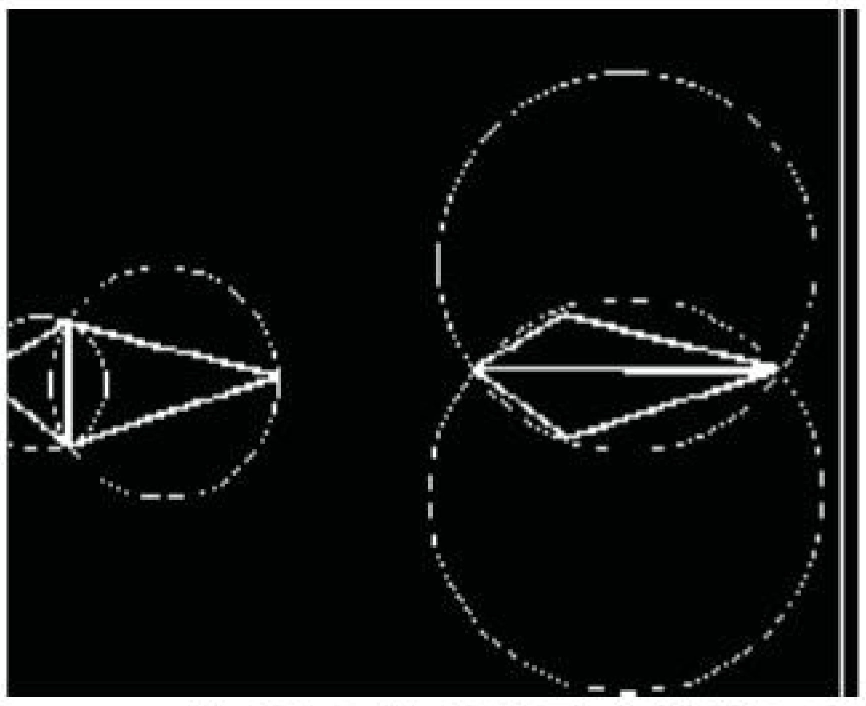


图 5.11.5：满足和不满足圆周测试的单元结构

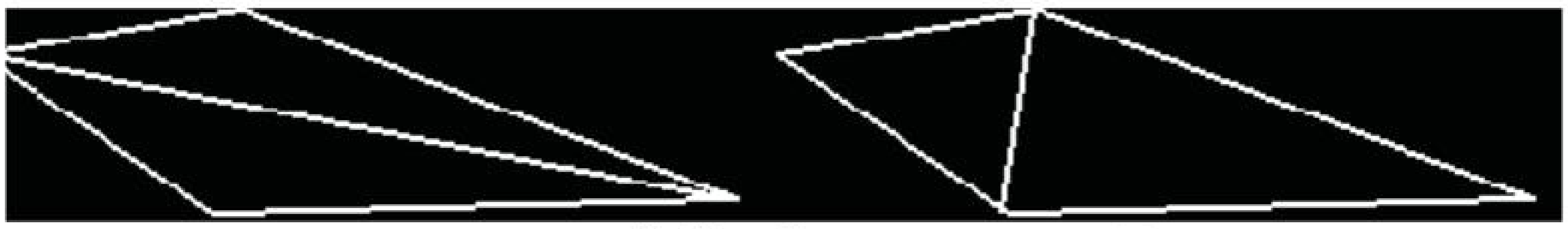


图 5.11.6：面被交换以满足 Delaunay 圆周测试

四面体网格

对于四面体网格，面的交换包括：寻找三个单元共享一个边的结构和把它们转换成两个单元共享一个面以降低歪斜和单元计数。（看图 5.11.7）

面交换的步骤

执行面交换，点击 **Swap** 按钮直到 **Number Swapped** 的报告为 0。**Number Visited** 表

示关于可能的面交换，被访问和测试的面的总数。

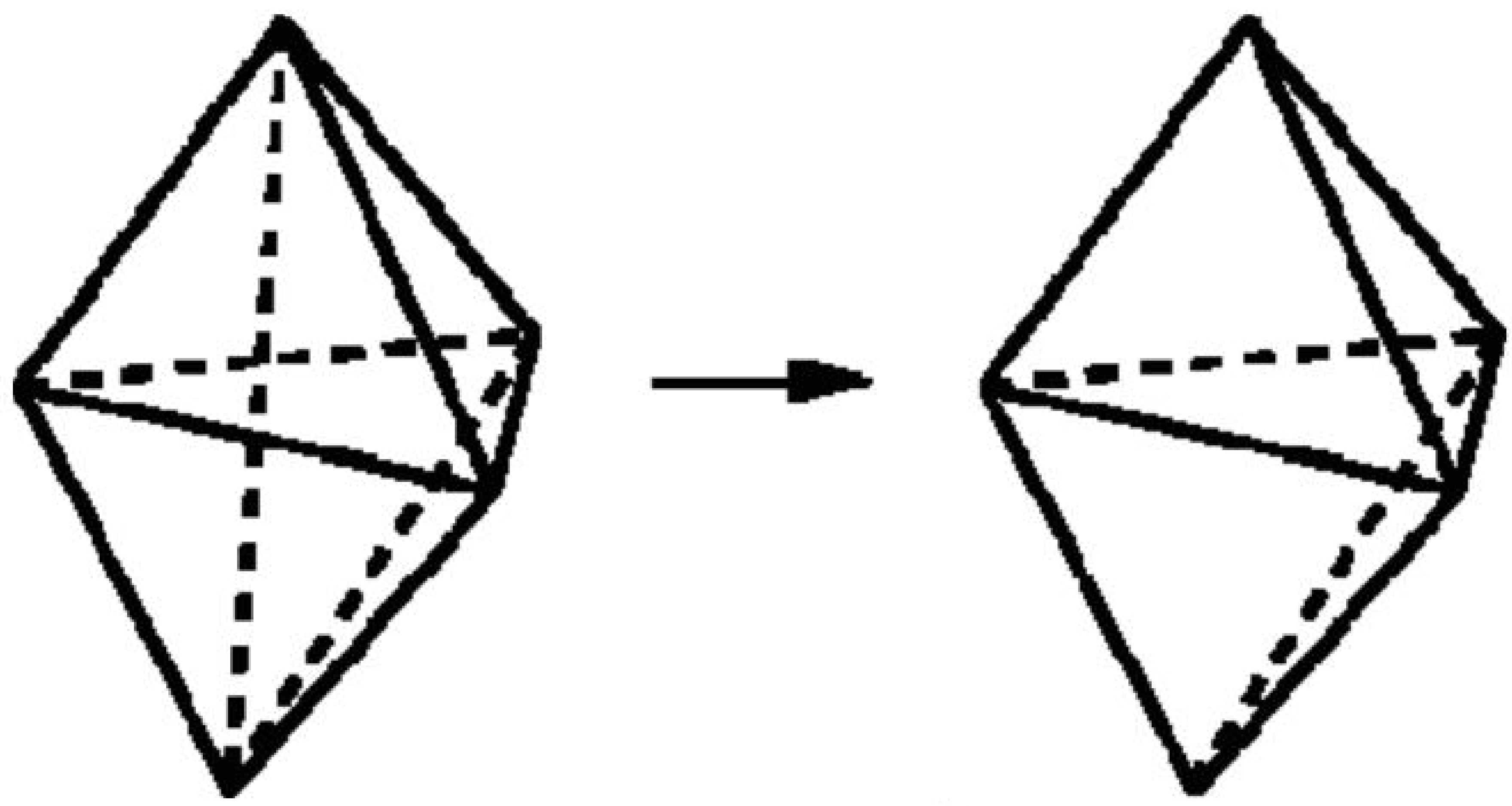


图 5.11.7：3D 面的交换

5.11.3 结合基于歪斜的光滑和面的交换

通常，基于歪斜的光滑应当与面的歪斜交替使用。关于这个程序的指定方法如下：

1．使用一个 Minimun Skewness－对于 3D 是 0.8，对于 2D 是 0.4－执行 4 个光滑迭代。

2．执行交换，直到 Number Swapped 降到 0。

3．对于 3D 网格，把 Minimum Skewness 减小到 0.6，重复光滑/交换的步骤。

第六章 求解技术

本章将讲述 FLUENT 的求解技术，包括数值格式、离散化方法、分离求解器、耦合求解器、多重网格法等内容，以及如何在计算中如何使用这些方法。求解技术是 CFD 软件的核心内容，了解这部分内容，对理解其他部分的内容有很大帮助。

6.1 数值格式回顾

FLUENT 中有两种求解器，即分离求解器（segregated solver）和耦合求解器（coupled solver）。这两种求解器的求解对象是相同的，即它们所求解的控制方程均为描述质量守恒、动量守恒和能量守恒的连续方程、动量方程和能量方程。在考虑湍流和化学反应时，还要加上湍流方程和化学组元方程。它们都用有限体积法作为对计算对象进行离散求解的基础方法，有限体积法的主要工作包括：

- 通过网格划分将空间区域分解成由离散的控制体组成的集合。
- 在控制体上用积分形式构造离散变量的代数方程。
- 将离散方程线化，然后通过求解线化方程获得变量的迭代解。

两种求解器的区别在于它们所使用的线化方法和求解离散方程的方法是不同的。

6.1.1 分离求解方法

分离求解方法即分别求解各个控制方程的方法。由于控制方程是非线性的，因此求解必须经过多次迭代才能获得收敛解。图 6-1 为求解流程图，其过程可概述如下：

（1）流场变量更新。在第一次计算时，变量由初始化过程更新。在随后的计算中，每迭代一次既得到一个更新的解。

（2）用当前压强和质量通量的值求解动量方程，以得到新的速度场。

（3）因为（2）中得到的速度场的数值解无法完全满足连续方程，于是再求解压强修正方程。压强修正方程是由连续方程导出的泊松型方程，求解这个方程可以得到对压强场、速度场和质量通量的修正，进而使连续方程得到满足。

（4）利用前面求出的解，求解湍流方程、能量方程、组元方程和能量方程。

（5）在多相流计算中如果考虑相间干扰，则需要通过求解弥散相轨迹计算得到连续相方程中的源项解。

（6）检验收敛条件是否被满足。如果收敛条件被满足，则停止计算。如果计算没有收敛，则继续迭代过程。

分离算法中采用压强速度耦合算法进行计算，具体格式包括 SIMPLE、SIMPLEC 和 PISO 三种。

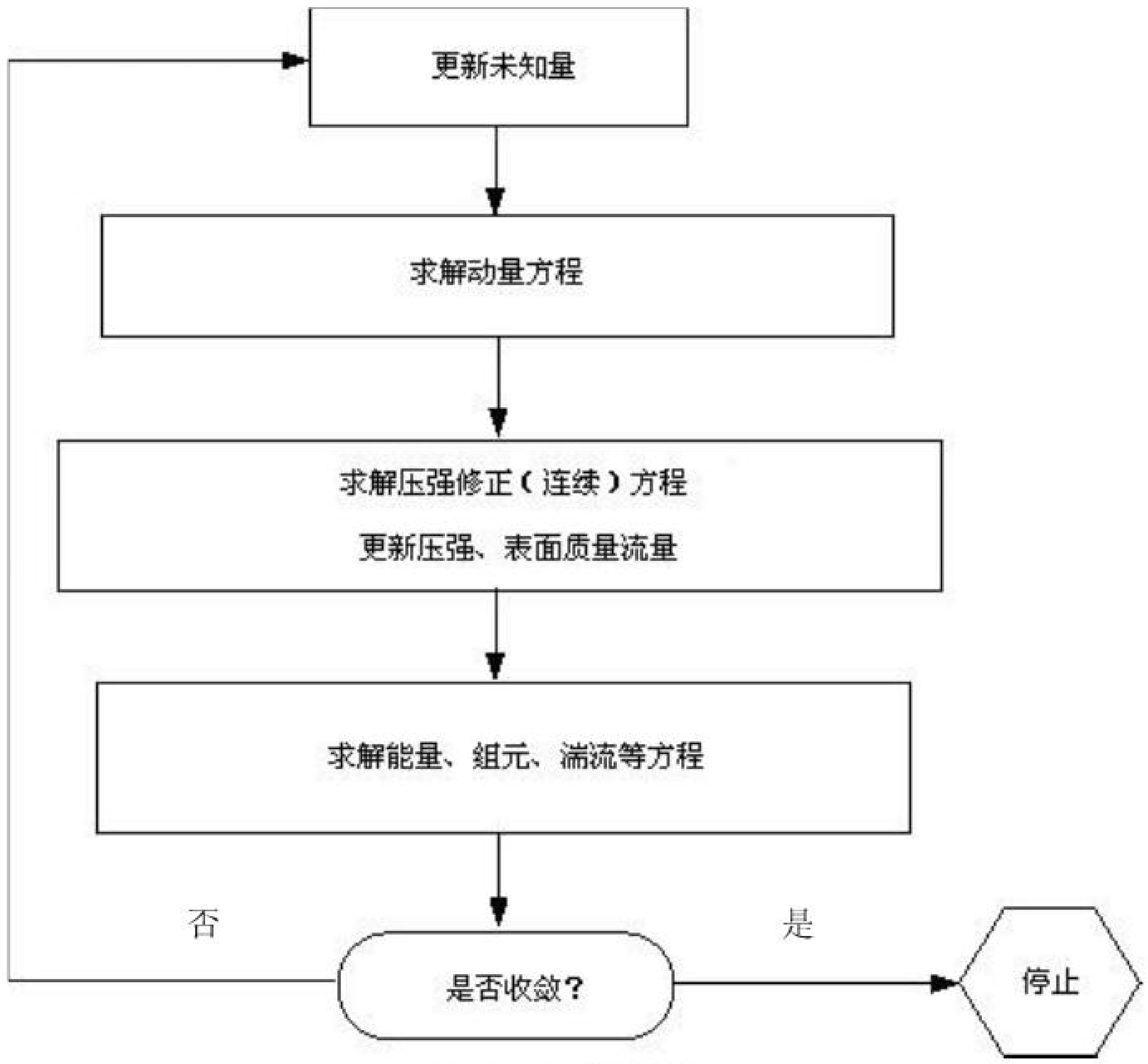


图 6-1 分离算法流程图

SIMPLE 算法的基本策略是用假定的压强场求解动量方程得到边界点上的通量。因为假定的压强场不准确，所以求得的通量必然不能满足连续方程，于是在通量上添加修正项，以使所得通量能够满足连续方程。而通量修正项是压强修正项的函数，因此将修正过的通量代入连续方程，就可以得到一个关于压强修正项的方程。用 AMG 多重网格法求解这个方程可以得到压强修正项的解。在压强修正项的前面乘以亚松弛因子，再与原压强相加就得到一个新的压强场。以这个新的压强场为起点重复上述过程，就形成交替求解压强场、速度场的迭代过程，直到最后得到收敛解，计算既告结束。

SIMPLEC 算法与 SIMPLE 算法的基本思路一致，仅在通量修正方法上有所改进，加快了计算的收敛速度。

SIMPLE 算法和 SIMPLEC 算法在每个迭代步中得到的压强场都不能完全满足动量方程，因此需要反复迭代，直到收敛。PISO 算法针对 SIMPLE 算法中每个迭代步获得的压强场与动量方程偏离过大的问题，在每个迭代步中增加了动量修正和网格畸变修正过程，因此虽然 PISO 在每个迭代步中的计算量大于 SIMPLE 算法和 SIMPLEC 算法，但是由于每个

迭代过程中获得的压强场更准确，所以使得计算收敛得更快，也就是说获得收敛解需要的迭代步数大大减少了。

6.1.2 耦合算法

与分离算法分别求解各个方程相反，耦合算法同时求解连续方程、动量方程和能量方程。在上述流场控制方程被求解后，再求解湍流、辐射等方程，所用方法与分离算法中相同。计算过程也需要经过迭代才能收敛得出最终的解。耦合算法的计算流程可以简单地用图 6-2 表示，大致过程为：

（1）流场变量更新。如果是第一步迭代，则流场变量由初始化过程更新。如果不是第一步计算，则由求解过程得到更新的解。

（2）同时求解连续方程、动量方程、能量方程和组元方程。

（3）在考虑湍流和辐射等模型的时候，求解湍流和辐射方程。

（4）在多相流计算中，如果考虑相间干扰，则通过弥散相轨迹计算求解连续相方程中的源项。

（5）检验计算是否收敛——如果计算已经收敛则停止计算，如果没有收敛则开始新一步的计算。

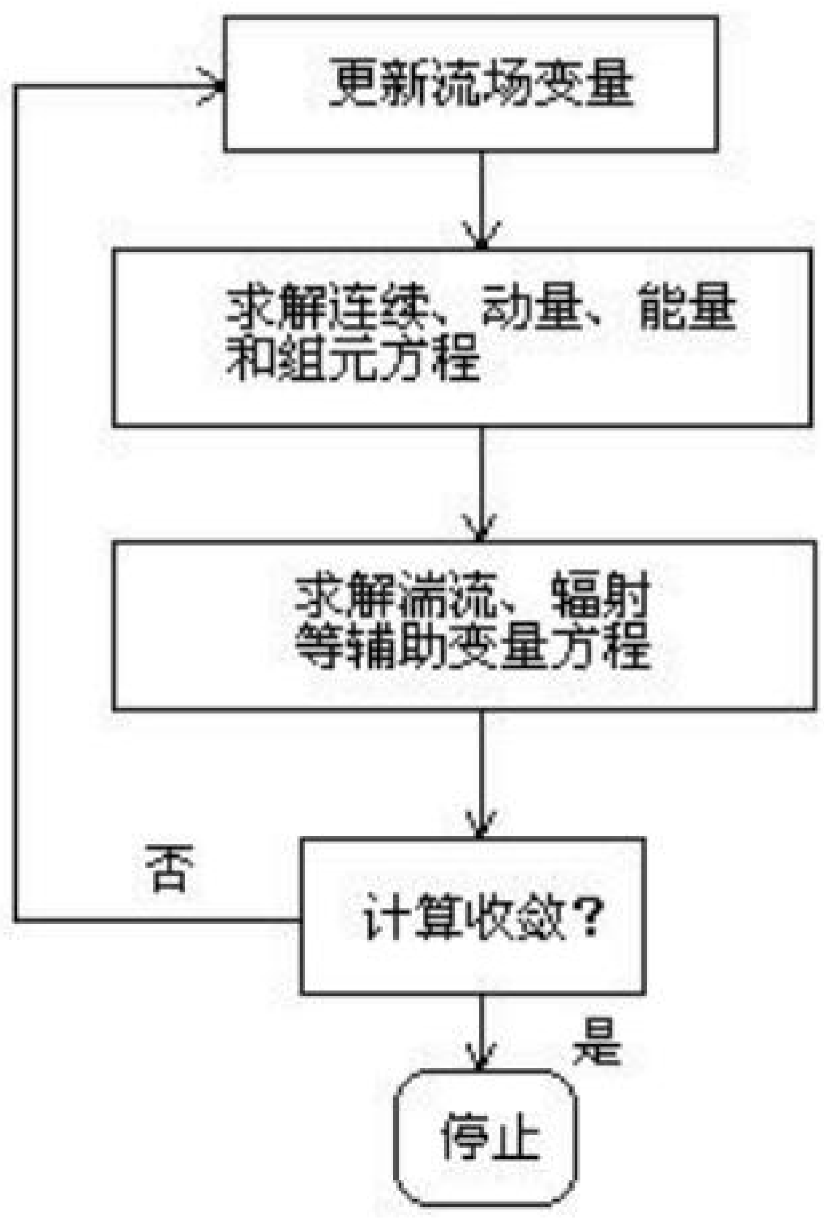


图 6-2 耦合算法流程图

6.1.3 方程的线化：隐式格式与显式格式的对比

流体的控制方程是非线性方程，在数值求解过程中需要将非线性方程在网格单元中化为线性方程，然后再进行求解。所谓隐式格式和显式格式是对方程进行线化和求解的两种不同方式。隐式格式将未知的流场变量（密度、速度、能量等）同已知量之间的关系用方程组的形式加以表达，然后通过求解方程组获得未知变量的值。显式格式则是将未知的流场变量写作已知量的显式函数形式，因此每个变量可以用一个方程单独进行求解。

分离算法中，每个流场变量是独立进行求解的，求解过程中采用隐式格式在各个网格点上单独进行求解。在 FLUENT 中，方程用 Gauss-Seidel 法与代数多重网格法（AMG）联合求解。

耦合算法中既可以用显式格式，也可以用隐式格式。在使用隐式格式时，耦合算法以求解方程组（而不是单独求解某个方程）的形式求出流场变量，因此方程求解使用块 Gauss-Seidel 法与 AMG 法联合完成。在使用显式格式时，方程组用 Runge-Kutta 法求解，同时用全逼近存储（Full Approximation Storage，FAS）多重网格法加速计算。

6.2 离散化

FLUENT 采用有限体积法将非线性偏微分方程转变为网格单元上的线性代数方程，然后通过求解线性方程组得出流场的解。网格划分可以将连续的空间划分为相互连接的网格单元。每个网格单元由位于几何中心的控制点和将网格单元包围起来的网格面或线构成。所谓求解流场控制方程，最终目的就是获得所有控制点上流场变量的值。

在有限体积法中，控制方程首先被写成守恒形式。从物理角度看，方程的守恒形式反映的是流场变量在网格单元上的守恒关系，即网格单元内某个流场变量的增量等于各边界面上变量的通量的总和。有限体积法的求解策略就是用边界面或线上的通量计算出控制点上的变量。比如对于密度场的计算，网格单元的控制点上的密度值及其增量代表的是整个网格单元空间上密度的值和增量。从质量守恒的角度来看，流入网格的质量与流出网格的质量应该等于网格内流体质量的增量，因此从质量守恒关系（连续方程）可以得知密度的增量等于边界面或线上密度通量的积分。

FLUENT 中用于计算通量的方法包括一阶迎风格式、指数律格式、二阶迎风格式、QUICK 格式、中心差分格式等形式，本节将分别进行介绍。

6.2.1 一阶迎风格式

“迎风”这个概念是相对于局部法向速度定义的。所谓迎风格式就是用上游变量的值计算本地的变量值。在使用一阶迎风格式时，边界面上的变量值被取为上游单元控制点上

的变量值。

6.2.2 指数律格式

指数律格式认为流场变量在网格单元中呈指数规律分布，在一维问题中这个指数规律如（6-1）式所示，这个公式由求解一维对流扩散方程得出。在对流作用起主导作用时，指数律格式等同于一阶迎风格式；在纯扩散问题中，对流速度接近于零，指数律格式等于线性插值，即网格内任意一点的值可以用网格边界上的值线性插值得到，而在对流和扩散作用相差无几时，需要用（6-1）式计算得出。

$$\frac{\phi(x)-\phi_0}{\phi_L-\phi_0}=\frac{\exp\left(Pe\frac{x}{L}\right)-1}{\exp(Pe)-1} \tag{6-1}$$

ϕ_0表示一维网格起点处的变量值，ϕ_L表示网格终点处的变量值，Pe为表征对流作用与扩散作用之比的 Peclet 数。对流作用强时，Pe 增大；反之，则 Pe 减小。

6.2.3 二阶迎风格式

一阶迎风格式和二阶迎风格式都可以看作流场变量在上游网格单元控制点展开后的特例：一阶迎风格式仅保留 Taylor 级数的第一项，因此认为本地单元边界点的值等于上游网格单元控制点上的值，其格式精度为一阶精度。二阶迎风格式则保留了 Taylor 级数的第一项和第二项，因而认为本地边界点的值等于上游网格控制点的值与一个增量[1]的和，因而其精度为二阶精度。

6.2.4 QUICK 格式

QUICK 格式用加权和插值的混合形式给出边界点上的值。QUICK 格式是针对结构网格，即二维问题中的四边形网格和三维问题中的六面体网格提出的，但是在 FLUENT 中，非结构网格计算也可以使用 QUICK 格式选项。在非机构网格计算中，如果选择 QUICK 格式，则非六面体（或四边形）边界点上的值是用二阶迎风格式计算的。在流动方向与网格划分方向一致时 QUICK 格式具有更高的精度。

[1] 这个增量等于流场变量的梯度与从控制点到边界点的矢径的点积。

6.2.5 中心差分格式

在使用 LES 湍流模型时，可以用二阶精度的中心差分格式计算动量方程，并得到精度更高的结果。

以本地网格单元的控制点为基点，对流场变量做 Taylor 级数展开并保留前两项，也可以得出边界点上具有二阶精度的流场变量值。在一般情况下，这样求出的边界点变量值与二阶迎风差分得到的变量值不同，而二者的算术平均值就是流场变量在边界点上的用中心差分格式计算出的值。

6.2.6 亚松弛方法

因为流场的控制方程是非线性的，所以 FLUENT 中采用亚松弛法控制流场变量的增量，即令经过迭代得到的增量略小于实际计算值：

$$\phi = \phi_{old} + \alpha\Delta\phi \qquad (6\text{-}3)$$

式中 ϕ 为实际计算结果，ϕ_{old} 为前一步的计算结果，α 为松弛因子，$\Delta\phi$ 为计算中得到的增量。

6.3 多重网格法

多重网格法是加速计算的有效手段，其基本原理是利用网格由细到粗，再由粗到细的循环过程，将流场的全局误差由低频误差转换为高频误差，从而可以利用现有格式可以有效降低高频误差的特点加速收敛，提高现有计算格式降低全局误差的速度，进而提高整个计算过程的计算速度。

FLUENT 中采用的 Gauss-Seidal 点隐格式和显式多步求解格式可以快速降低局部误差（高频误差），但是因为这些格式是以控制点为中心的，所以在降低全局误差（低频误差）方面效率很低，用于降低全局误差的计算步数通常比降低局部误差的计算步数高出 2～3 个数量级，因此加速全局误差的降低速度就可以大大加快流场计算的收敛速度。

FLUENT 中采用的多重网格法包括 AMG（Algebraic Multi-Grid，代数多重网格法）法和 FAS（Full-Approximation Storage，全逼近存储法）两种方法。其中 AMG 方法是 FLUENT 中用于多重网格计算的基本算法，既可以与分离求解器配合使用，也可以与耦合求解器配合使用；FAS 方法在计算非线性问题方面优于 AMG 方法，但是只能与耦合显式求解器配合使用，同时因为 FAS 方法在计算过程中需要存储每一层网格，所以消耗的计算资源比较多，建议在计算机配置比较高时采用。

用多重网格法进行计算时，由细网格到粗网格的变化过程被称为约束（Restriction），由粗网格到细网格的变化过程被称为扩张（Prolongation）。在这两个过程中，离散方程的形式会相应发生变化。每遍历一次由细到粗，再由粗到细的过程被称为一个循环，根据循环过程的不同，又可以进一步将循环类型划分为 V 循环、W 循环、Flex（灵活）循环，以及由 V 循环和 W 循环为基本结构组成的 F 循环。V 循环的基本特征是在一个循环过程中只计算一次粗网格流场，形象地说是只有一次“触底”过程，而 W 循环则有二次触底过程，F 循环则由一次 W 循环和紧跟在后面的一次 V 循环组成。Flex 循环是根据实际计算情况灵活调整循环设置的循环类型，即在收敛速度很慢时访问粗网格层以加速收敛，在粗网格层上计算收敛后再返回细网格层进行计算。

多重网格计算中究竟设置多少层网格在理论上没有限制，就大多数情况而言，3~5 个网格层就足够了。粗网格的构建方法对于结构网格和非结构网格是不同的。对于结构网格，只要以细网格为基础，去掉一些网格分隔线就可以达到粗化网格的目的，同时计算中的约束和扩张过程也比较容易通过线性插值的形式完成，比如一个“田”字形的网格可以通过去掉中间的网格分隔线变成“口”字形的粗网格。而非结构网格中粗网格的生成方法比较复杂。在 AMG 方法中，粗网格是通过将流场变量关联较强的相邻网格合并得到的。在 FAS 方法中，粗网格是通过将节点附近的所有网格融合为一个网格得到的。

6.4 使用求解器的基本步骤

使用求解器的基本步骤如下：

（1）选择计算格式。在使用分离求解器时，选择压强插值格式。

（2）在使用分离求解器时，选择压强和速度的耦合方法。

（3）在使用分离求解器时，选择多孔介质速度方法。

（4）选择用于计算导数的梯度选项。

（5）设置亚松弛因子。

（6）在使用耦合求解器时，选择使用 FAS 多重网格法。

（7）对所选用的模型和求解器进行细节设置。

（8）对流场进行初始化。

（9）选择求解过程监视器。

（10）开始迭代计算。

（11）如果计算不收敛，尝试本章最后一节中关于改善收敛性的方法。

在随后的几节中将对上面步骤中的重点内容进行介绍。

6.5 选择离散格式

在计算之前可以为对流项的计算选择离散格式，粘性项则自动采用二阶格式。在缺省情况下，分离求解器用一阶迎风格式计算所有方程的对流项，耦合求解器则用二阶格式计算连续、动量和能量方程等主控方程，仅用一阶格式计算湍流和辐射等辅助方程。在选用分离求解器时可以选择压强插值格式。

6.5.1 计算格式的选择

对流项的计算格式

在流动方向与网格方向基本一致时，可以选择一阶迎风格式。一阶迎风格式适用于结构网格，具有稳定性高，计算速度快的优点，但是在网格方向与流动方向不一致时，产生的数值误差比较大。在非结构网格计算，或流场比较复杂的情况下，很难保证流动方向与网格相一致，此时应该选用二阶格式。

二阶格式的计算精度高于一阶格式，但是相对而言，其计算时间比较长，收敛性也相对较差，因此在实际计算中进行格式选择时，需要兼顾精度、收敛性和系统资源等方面的要求。比如可以在计算开始时先用一阶格式进行计算以获得一个相对粗糙的解，在计算收敛后再用二阶格式完成计算以提高解的精度。这样既避免了二阶格式收敛性差、计算时间长的问题，又避免了一阶格式在复杂流场计算中数值误差大的问题。除此之外，在采用二阶格式无法收敛的情况下，也可以考虑改用一阶格式进行计算。

在用结构网格计算旋转流动问题时，QUICK 格式可以提供更高的计算精度，但是在其他情况下 QUICK 格式的精度与二阶格式相当，并没有很大的改进。计算中还可以选用指数律格式，其精度与一阶格式基本相同。中心差分仅在使用 LES 湍流模型时可以使用，并且应该在网格足够密集、局部 Peclet 数小于 1 的情况下使用。

压强插值格式的选择

在大多数情况下，可以使用 FLUENT 缺省设置的“标准”格式，对于下列特殊情况可以分别选择相应的格式：

- 在彻体力对流场有很大影响的情况下，应该选择彻体力加权（body-force-weighted）格式。
- 在流场中有涡量很大的集中涡、高雷诺数自然对流、高速旋转流、多孔介质，以及流线曲率很大时，应该选择 PRESTO!格式。
- 对于可压流，应该使用二阶格式。
- 二阶格式不能用于多孔介质计算和多相流计算中的混合物模型及 VOF 模型。在其他情况下，为了提高精度可以选用二阶格式。

密度插值格式

在用分离算法计算单相可压流时，有三种密度插值格式可供选择，即一阶迎风格式、二阶格式和 QUICK 格式。一阶迎风格式具有良好的稳定性，但是在计算带激波的可压流时，会对激波解产生“抹平”作用，因此应该选用二阶格式或 QUICK 格式。在用四边形网格、六面体网格或混合网格计算带激波的流动时，最好使用 QUICK 格式计算所有变量。需要注意的是，在计算可压多项流时，只能用一阶迎风格式计算可压缩相的流动。

6.5.2 软件中的实际操作过程

在 FLUENT 软件中选择计算格式是在 Solution Controls（求解过程控制）面板中完成的，如图 6-3 所示。进入这个面板的菜单操作步骤如下：

Solve->Controls->Solution...



图 6-3 求解过程控制面板

在 Discretization（离散）组中，可以在使用分离求解器时，定义动量、能量、湍流动能等项目，并为这些项目选择的一阶迎风格式、二阶迎风格式、指数律格式、QUICK 格式和中心差分格式（在 LES 湍流模式计算中），也可以在使用耦合求解器时，定义湍流动能、湍流耗散率等项目，并为这些项目选择一阶迎风格式、二阶迎风格式。

需要注意的是，图 6-3 是选择分离算法时 Solution Controls（求解过程控制）面板的外

观，采用耦合算法时这个面板的外观有所不同。

在使用分离算法时，可以在 Pressure（压强）旁边的列表中选择压强插值格式，其中包括标准格式、线性格式、二阶格式、彻体力加权格式和 PRESTO!格式。在计算可压流时，需要在 Density（密度）旁边的列表中选择密度插值格式，包括一阶迎风、二阶迎风和 QUICK 格式。

按 Default（缺省）按钮可以将所有设置恢复为缺省设置。

6.6 选择压强—速度关联算法

在使用分离求解器时，可以选择三种压强—速度的关联形式，即 SIMPLE、SIMPLEC 和 PISO。SIMPLE 和 SIMPLEC 通常用于定常流计算，PISO 用于非定常计算，但是在网格畸变很大时也可以使用 PISO 格式。

6.6.1 格式选择

FLUENT 缺省设定的格式为 SIMPLE 格式，但是因为 SIMPLEC 稳定性较好，在计算中可以将亚松弛因子适当放大，所以在很多情况下可以考虑选用 SIMPLEC。特别是在层流计算时，如果没有在计算中使用辐射模型等辅助方程，用 SIMPLEC 可以大大加速计算速度。在复杂流动计算中，二者收敛速度相差不多。

PISO 格式通常被用于非定常计算，但是它也可以用于定常计算。PISO 格式允许使用较大的时间步长进行计算，因而在允许使用大时间步长的计算中可以缩短计算时间。但是在类似于大涡模拟（LES）这类网格划分较密集，因而时间步长很小的计算中，采用 PISO 格式计算则会大大延长计算时间。另外在定常问题的计算中，PISO 格式与 SIMPLE 和 SIMPLEC 格式相比并无速度优势。

PISO 格式的另一个优势是可以处理网格畸变较大的问题。如果在 PISO 格式中使用邻近修正[2]（neighbor correction），可以将亚松弛因子设为 1.0 或接近于 1.0 的值。而在使用畸变修正（skewness correction）时，则应该将动量和压强的亚松弛因子之和设为 1.0，比如将压强的亚松弛因子设为 0.3，将动量的亚松弛因子设为 0.7。如果同时采用两种修正形式，则应将所有松弛因子设为 1.0 或接近于 1.0 的值。

在大多数情况下都不必修改缺省设置，而在有严重网格畸变时，可以解除邻近修正和畸变修正之间的关联关系。

[2]即对动量进行修正

6.6.2 在软件中设置格式

在 Solution Controls（求解过程控制）面板中（见图 6-3），在 Discretization（离散）组下面的压强速度耦合列表中选择 SIMPLE、SIMPLEC 或 PISO 格式即可。

如果选择 PISO，则面板会自动展开出现 PISO 参数设置栏。在缺省情况下，畸变修正（Skewness Correction）和邻近修正（Neighbor Correction）的值都是 1。如果仅使用畸变修正，则将邻近修正的值设为 0，反之，则将畸变修正的值设为 0。缺省情况下，Skewness-Neighbor Coupling（畸变邻近关联）选项是被选中的，可以根据情况取消二者的关联关系。

6.7 设置亚松弛因子

如前所述，FLUENT 中各流场变量的迭代都由亚松弛因子控制，因此计算的稳定性与亚松弛因子紧密相关。在大多数情况下，可以不必修改亚松弛因子的缺省设置，因为这些缺省值是根据各种算法的特点优化得出的。在某些复杂流动情况下，缺省设置不能满足稳定性要求，计算过程中可能出现振荡、发散等情况，此时需要适当减小亚松弛因子的值，以保证计算收敛。

在实际计算中可以用缺省设置先计算 4、5 步，如果发现残差曲线向上发展，则中断计算，适当调整亚松弛因子后再继续计算。在修改计算控制参数前，应该先保存当前计算结果。调整参数后，计算需要经过几步调整才能适应新的参数。一般而言，增加亚松弛因子将使残差增加，但是如果格式是稳定的，增加的残差仍然会逐渐降低。如果改变参数后，残差增加了几个量级，就可以考虑中断计算，并重新调入保存过的结果，再做新的调整。

在计算发散时，可以考虑将压强、动量、湍流动能和湍流耗散率的亚松弛因子的缺省值分别降低为 0.2、0.5、0.5、0.5。在计算格式为 SIMPLEC 时，通常没有必要降低亚松弛因子。

亚松弛因子是在 Solution Controls（求解过程控制）面板中 Under-Relaxation Factors（亚松弛因子）旁的输入栏中设定的。点击 Default（缺省）按钮可以恢复缺省设置。

6.8 改变库朗数

FLUENT 中的时间步长是由库朗数定义的，而库朗数是由线性稳定性理论定义的一个范围，在这个范围内计算格式是稳定的。给定一个库朗数，就可以相应地得到一个时间步长。库朗数越大，时间步长就越长，计算收敛速度就越快，因此在计算中库朗数都在允许的范围内尽量取最大值。

6.8.1 耦合显式格式中的库朗数

在显式格式和隐式格式中库朗数的取值范围差别很大，在显式格式中库朗数的取值范围很小，隐式格式则宽松得多。

在 FLUENT 使用的多步格式中，可以认为库朗数在小于 2.5 的范围内是稳定的。由于控制方程是非线性的，所以库朗数的取值一般达不到线性稳定性分析得到的极限值。在耦合算法的显式格式中，系统设定的缺省值为 1.0，在某些二维问题中可以适当放大这个值，但是不要超过 2.0。

如果计算模型的设定是正确的，并且进行了初始化，而计算中却发现残差快速上升，通常说明库朗数的值可能需要降低。另外，在计算的开始阶段，因为初始流场相对粗糙，所以可以适当降低库朗数，比如降低到 0.1－0.5 左右，然后在计算相对稳定后再适当调高库朗数。

6.8.2 耦合隐式格式中的库朗数

在线性稳定性理论表明 FLUENT 中使用 Gauss-Seidel 格式是无条件稳定的，但是由于控制方程非线性的影响，实际上库朗数在这种情况下也不可能取为无限大。在隐式格式中，库朗数的缺省值为 5.0，在很多情况下可以将缺省值改为 10、20、100 甚至更高，具体取多高主要取决于问题的复杂程度。与显式格式一样，在计算开始的时候可以把库朗数取得小一些，而在经过几个迭代步后可以将库朗数再调高。

AMG 求解器可以在多重网格的计算过程中发现计算是否已经发散。如果计算开始发散，AMG 求解器会自动降低库朗数并重新进行迭代，同时屏幕上会显示相关信息。在迭代成功后，库朗数会恢复原来的值，迭代计算则继续进行下去。如果迭代不成功，计算就会中断。

6.8.3 库朗数在 FLUENT 中的设定

库朗数是在 Solution Controls（求解过程控制）面板中设定的。在 Solver Parameters（求解器参数）组中的 Courant Number（库朗数）旁边的栏中输入其值就可以了。如果选择的求解器是显式格式，则系统会自动将库朗数设为缺省值 1。如果选择的求解器是隐式格式，则系统设定的缺省值为 5。

6.9 引入 FAS 多重网格

如前面章节所述，FAS 多重网格是耦合算法显式格式的一个可选项。因为几乎所有耦

合显式格式计算都可以在使用 FAS 多重网格法时得到加速，所以在计算中应该尽量采用 FAS 多重网格法。采用多重网格法唯一需要设定的参数是网格的级数，只要将这个参数设置为不等于零的数，系统就自动在计算中启用 FAS 多重网格法。

需要注意的是，AMG 代数多重网格法在缺省设置中是永远被使用的方法。另外，FAS 多重网格法不能与显式时间步长配合使用，因为粗网格计算将打乱细网格计算中对时间步长的计算。

如前所述，FAS 多重网格法在求解过程中将粗网格的解传递回细网格，因此可以加快流场变量信息的传播速度，并加快计算收敛进程。控制多重网格求解过程的基本办法是设置粗网格的级数。在多重网格法中，虽然粗网格的级数越多，收敛就越快，但是也不能无限制增加粗网格级数，因为级数越多，占用系统资源也越多。同时粗网格是由临近的细网格结合而成的，理论上存在一个极限，即在某个级数上可能其次级“细”网格仅有一个网格单元，此时再增加网格级数已无意义，FLUENT 在这种情况下会假定高级网格等同于其次级网格，而不再增加网格级数。

在 FLUENT 软件中可以在两个地方设置网格级数，即 Solution Controls（求解控制）面板和 Multigrid Controls（多重网格控制）面板，Multigrid Controls（多重网格控制）面板可以用下面的菜单操作进入：

Solve->Controls->Multigrid...

改变粗网格级数，系统就会自动更新内部设置，启用 FAS 多重网格法。

6.10 设置求解极限

流场变量在计算过程中的最大、最小值可以在 Solution Limits（求解极限）面板中设定，菜单操作次序为：

Solve->Controls->Limits...

设置解变量极限是为了避免在计算中出现非物理解，比如密度或温度变成负值，或者大得远远超过真实值。FLUENT 中同时可以对温度的变化率极限进行设置，这样可以避免因为温度变化过于剧烈而导致温度出现负值，温度变化率的缺省设置是 0.2，即温度变化率不能超过 20%。

在计算之前可以对缺省设定的解变量极限进行修改，比如温度的缺省设置是 5000K，但是在一些高温问题的计算中，可以将这个值修改为更高的值。另外，如果计算过程中解变量超过极限值，系统会在屏幕上发出提示信息，提示在哪个计算区域、有多少网格单元的解变量超过极限。对湍流变量的限制是为了防止湍流变量过大，对流场造成过大的、非物理的耗散作用。

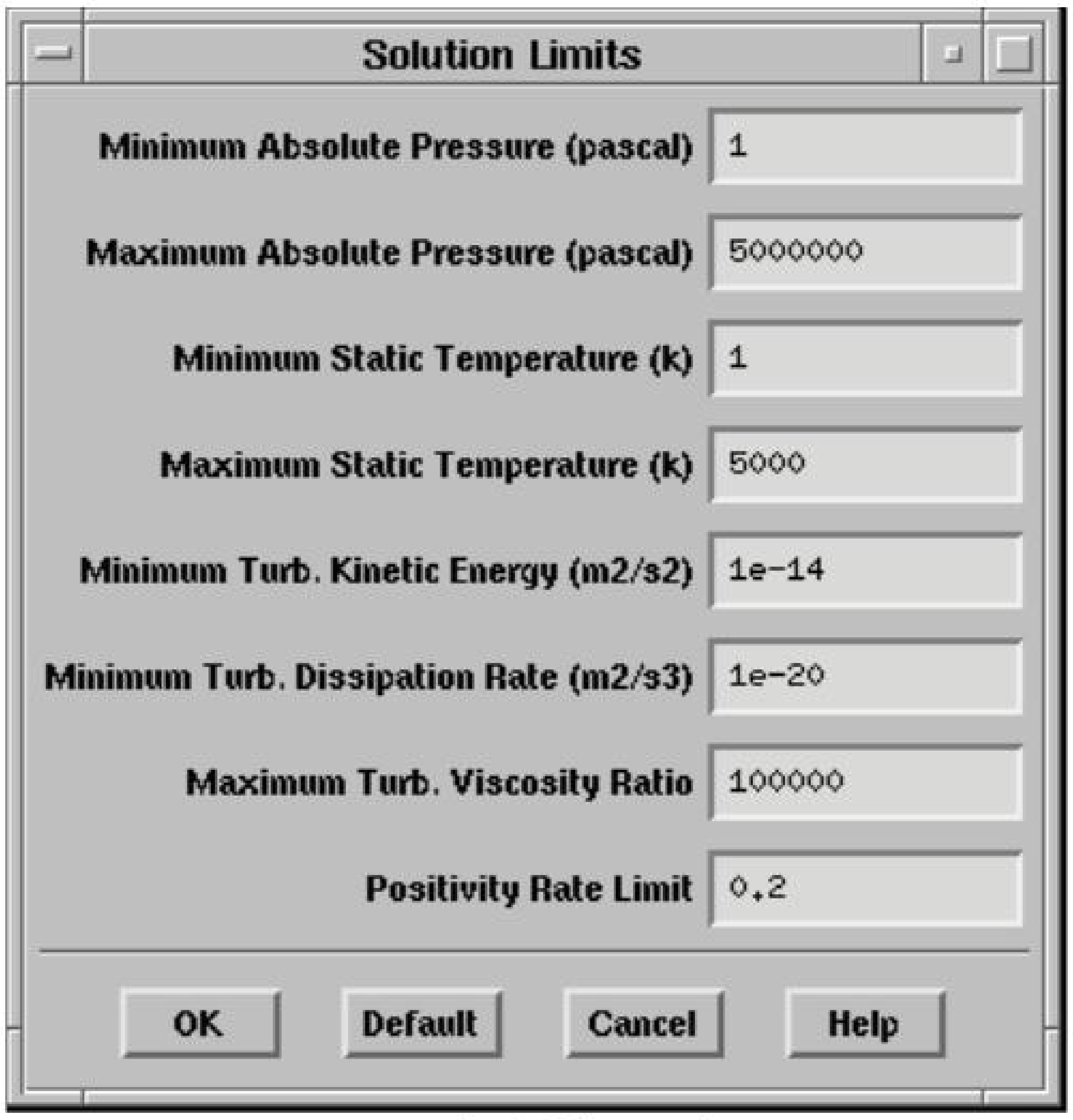


图 6-4 解变量极限面板

6.11 初始化

在开始进行计算之前，必须为流场设定一个初始值。设定初始值的过程被称为“初始化”。如果把每步迭代得到的流场解按次序排列成一个数列，则初始值就是这个数列中的第一个数，而达到收敛条件的解则是最后一个数。显然如果初始值比较靠近最后的收敛解，则会加快计算过程，反之则会增加迭代步数，使计算过程加长，更严重的是如果初始值给的不好，有可能得不到收敛解。

在 FLUENT 中初始化得方法有两种，即

（1）全局初始化，即对全部网格单元上的流场变量进行初始值设置。

（2）对流场进行局部修补，即在局部网格上对流场变量进行修改。

在进行局部修补之前，应该先进行全局初始化。

6.11.1 全局初始化

计算前对流场进行的全局初始化是在 Solution Initialization（初始化）面板中进行设置的。用下列次序操作可以进入这个面板：

Solve->Initialize->Initialize...

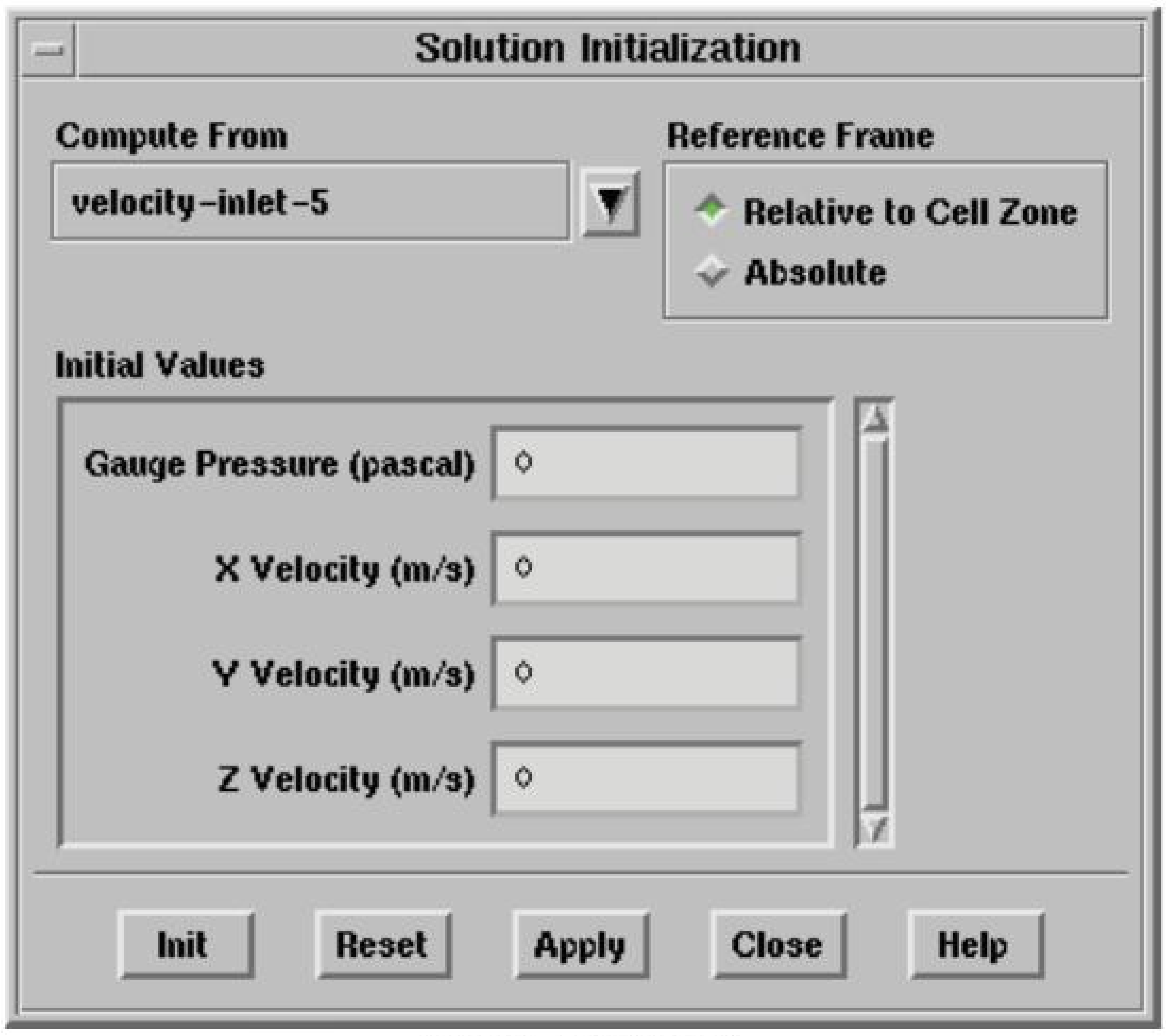


图 6-5 初始化面板

初始化的步骤如下：

（1）设定初始值：

- □ 如果想用某个区域上设定的初始值进行全局初始化，应该先在 Compute From（计算起始位置）列表中选择需要定义初始值的区域名，再在 Initial Values（初始值）中给定各变量的值，则所有流场区域的变量的值都会根据给定区域的初始值完成初始化过程。
- □ 如果用平均值的办法对流场进行初始化，则在 Compute From 列表中选择 all-zones（所有区域），则 FLUENT 将根据边界上设定的值计算出初始值，完成对流场的初始化。
- □ 如果希望对某个变量的值做出改变，可以直接在相应的栏目中输入新的变量值。

（2）如果计算中使用了动网格，可以通过选择“Absolute（绝对速度）”或“Relative to Cell Zone（相对于网格区域）”来决定，设定的初始值为绝对速度值，还是相对速度。缺省设置为相对速度。

（3）在检查过所有初始值的设定后，可以点击 Init（初始化）按钮开始流场的初始化。如果初始化是在计算过程中重新开始的，则必须用 OK 键确认用新的初始值覆盖计算值。

初始化面板下面几个按钮的含义为：

- □ Init（初始化）按钮，保存初始值设置，并进行初始化计算。
- □ Apply（应用）按钮，仅进行初始值保存操作。
- □ Reset（重置）按钮，如果初始化过程有错误，比如初始值有错误，或者使用了错

误的区域作为开始区域，则可以按此键将初始值恢复为缺省值。

6.11.2 对初始值进行局部修补

在完成全局初始化后，可能会对某些变量的值进行局部修补。局部修补在修补（Patch）面板中进行设定。

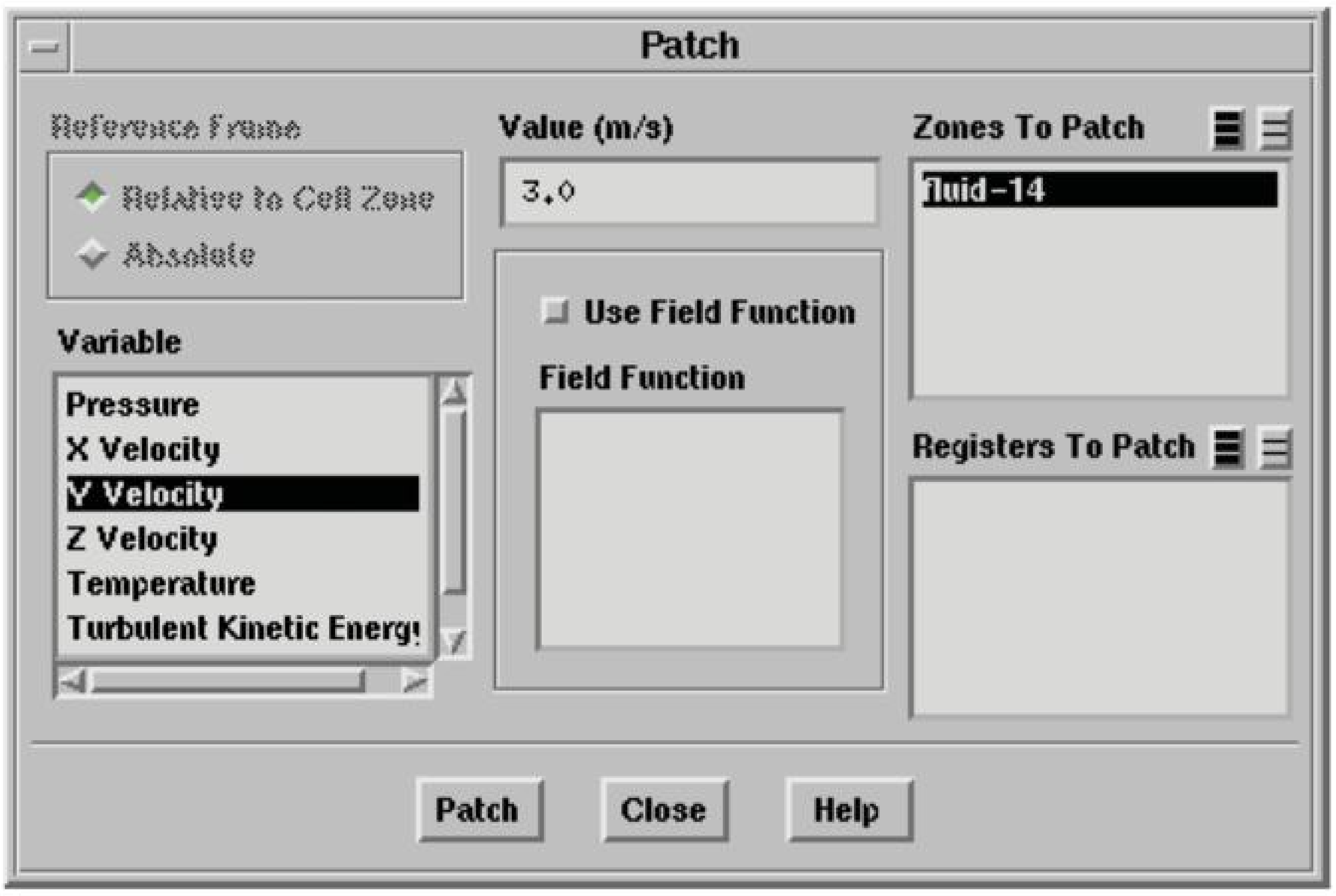


图 6-6 修补面板

局部修补的步骤如下：

（1）在 Variable（变量）列表中选择需要修补的变量名。

（2）在 Zones To Patch（需要修补的区域）或 Registers To Patch（需要修补的标记区）中选择需要修补变量所在的区域。

（3）如果需要将变量的值修补为常数，则直接在输入栏中输入变量的值。如果需要用一个预先设定的函数定义变量，可以在 Use Field Function（使用场变量函数）中的 Field Function（场函数）列表中选择合适的场函数。

（4）如果需要修补的变量为速度，则除了定义速度的大小，还要定义速度为绝对速度，还是相对速度。

（5）点击 Patch（修补）按钮更新流场数据。

局部修补通常是针对某个流场区域进行的，而用标记区进行局部修补则可以对某个流场区域中的一部分网格上的变量值进行修补。标记区可以用网格的物理坐标、网格的体积特征、变量的梯度或其他参数进行标记。在创建了标记区后，就可以对标记区上的初始值进行局部修补操作了。

用 Custom Field Function Calculator（编制场函数算子）面板可以编制自己的场函数，然后用场函数来反映物理量在流场中的变化过程。

因为局部修补不影响流场的其他变量，所以也可以在计算过程中，用局部修补的方法改变某些变量的值，对计算过程进行人为干预。

6.12 流场求解

6.12.1 定常问题计算

定常状态的计算在 Iterate（迭代）面板中开始，迭代面板的开启过程如下：

Solve->Iterate...

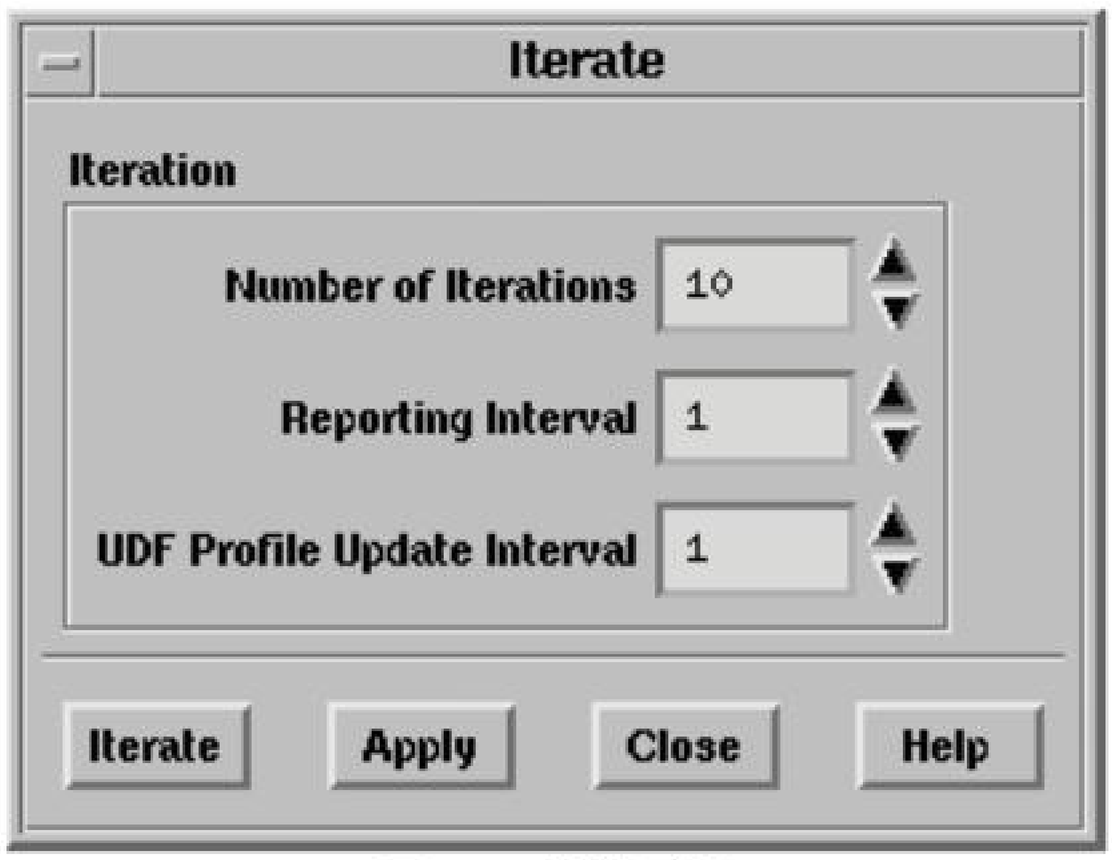


图 6-7 迭代面板

迭代面板中第一个输入栏为 Number of Iterations（迭代步数），在这里填入计算需要迭代的步数。第二栏为 Reporting Interval（报告间隔），即每隔多少步显示一次求解信息，缺省设置为 1。如果计算中使用了 UDF 函数，则可以用第三行中的输入栏决定每隔多少步输出一次 UDF 函数的更新信息。

设置完毕后，点击 Iterate（迭代）按钮就可以开始计算了。在计算开始后，会弹出一个工作窗口提示迭代正在进行，如果想中断计算，可以点击窗口中的 Cancel（取消）按钮就可以中断计算。如果想继续计算，则再次点击 Iterate 按钮就可以了。如果对计算结果不满意，希望重新开始计算的话，则要重新初始化一次。

6.12.2 时间相关问题的计算

用户输入

FLUENT 可以模拟的时间相关问题包括旋涡卷动、可压缩振荡、瞬态热传导、瞬态化学反应与混合流动等问题，比如圆柱后面卡门涡街中旋涡的生成和脱落过程等等。

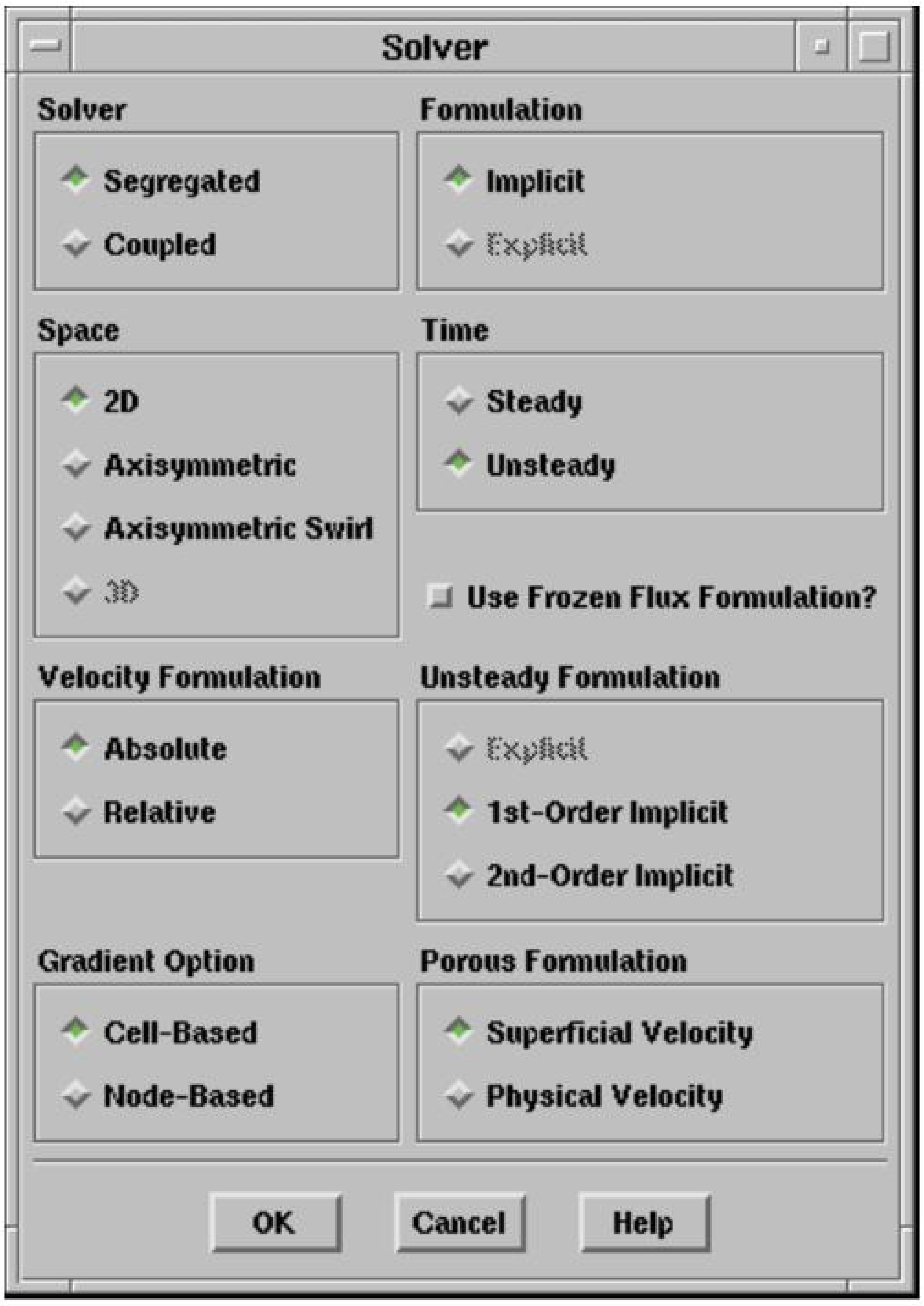


图 6-8 求解器面板

求解时间相关问题时，求解器的设置步骤如下：

（1）求解器面板的启动步骤为：

Define->Models->Solver...

首先选中 Unsteady（非定常）选项，然后再在 Unsteady Formulation（非定常算法）中选择所需要的算法。一般地说，1st-Order Implicit（一阶隐式格式）适用于大多数情况。如果要提高精度，可以使用 2nd-Order Implicit（二阶隐式格式）。在选择耦合显式求解器时，

可以使用 Explicit（显式）格式。显式格式主要用于激波等波动解的捕捉问题。

在使用分离求解器时，还可以选择 Use Frozen Flux Formulation（使用冻结通量格式）选项，但是这个选项只能在用分离求解器计算单相流问题时使用，同时计算中还不能使用动网格技术。

（2）定义相关物理模型和边界条件。

（3）如果选用了分离求解器，则应该在压强－速度关联方法中选择 PISO 格式。在用 LES 湍流模型计算湍流问题时，不应该选择 PISO 格式，那样会增加系统资源消耗。

（4）如果使用的是显式非定常计算或适应性时间推进法，应该在每步迭代中打开当前时间或当前时间步长的 print 选项。这项操作在 Statistic Monitors（统计数据监视器）面板中完成，通过下列菜单操作可以进入这个面板：

Solve->Monitors->Statistic...

在 Statistics（统计数据）列表中选择 time（时间）或 delta_time（时间增量），并打开 Print 选项。当 FLUENT 在控制台窗口打印残差信息时，就会出现当前时间或当前时间步长。

（5）可以用 Force Monitors（受力监视器）或 Surface Monitors（表面监视器）面板监视随时间变化的受力系数的值或平均值、质量平均值、积分或场变量通量，或者任意一个函数在表面上随时间变化的情况。

（6）用 Solution Initialization（求解初始化）面板设定初始条件：

Solve->Initialize->Initialize...

还可以读入一个定常状态的数据文件来设定初始条件：

File->Read->Data...

（7）用自动保存功能定义文件名和在迭代过程中自动保存算例文件和数据文件的频率：

File->Write->Autosave...

还可以用 Execute Commands（执行命令）面板自动执行系统命令。

（8）如果想创建解的动画过程，可以使用 Solution Animation（求解动画）面板设置动画显示参数。

（9）监视时间数据的步骤如下：

1）在 Iterate（迭代）面板中打开 Data Sampling for Time Statistics（时间数据采样）选项：

Solve->Iterate...

打开这个选项可以同时显式平均值和均方根值。

2）初始化流场统计数据：

Solve->Initialize->Reset Statistics

用这个菜单还可以在收集了一些数据后重置流场数据。比如在将时间采样选项打开的情况下先计算 10 步，检查计算结果，然后再计算 10 步，则时间统计数据中将包括前面 10 步的数据，除非重新对流场统计数据进行初始化。

（10）指定时间相关解的参数并开始计算，对于定常、非定常计算的步骤如下：

1）如果选择一阶或二阶精度隐式计算，步骤如下：

□ 在 Iterate（迭代）面板中定义时间相关解的参数：

Solve->Iterate...

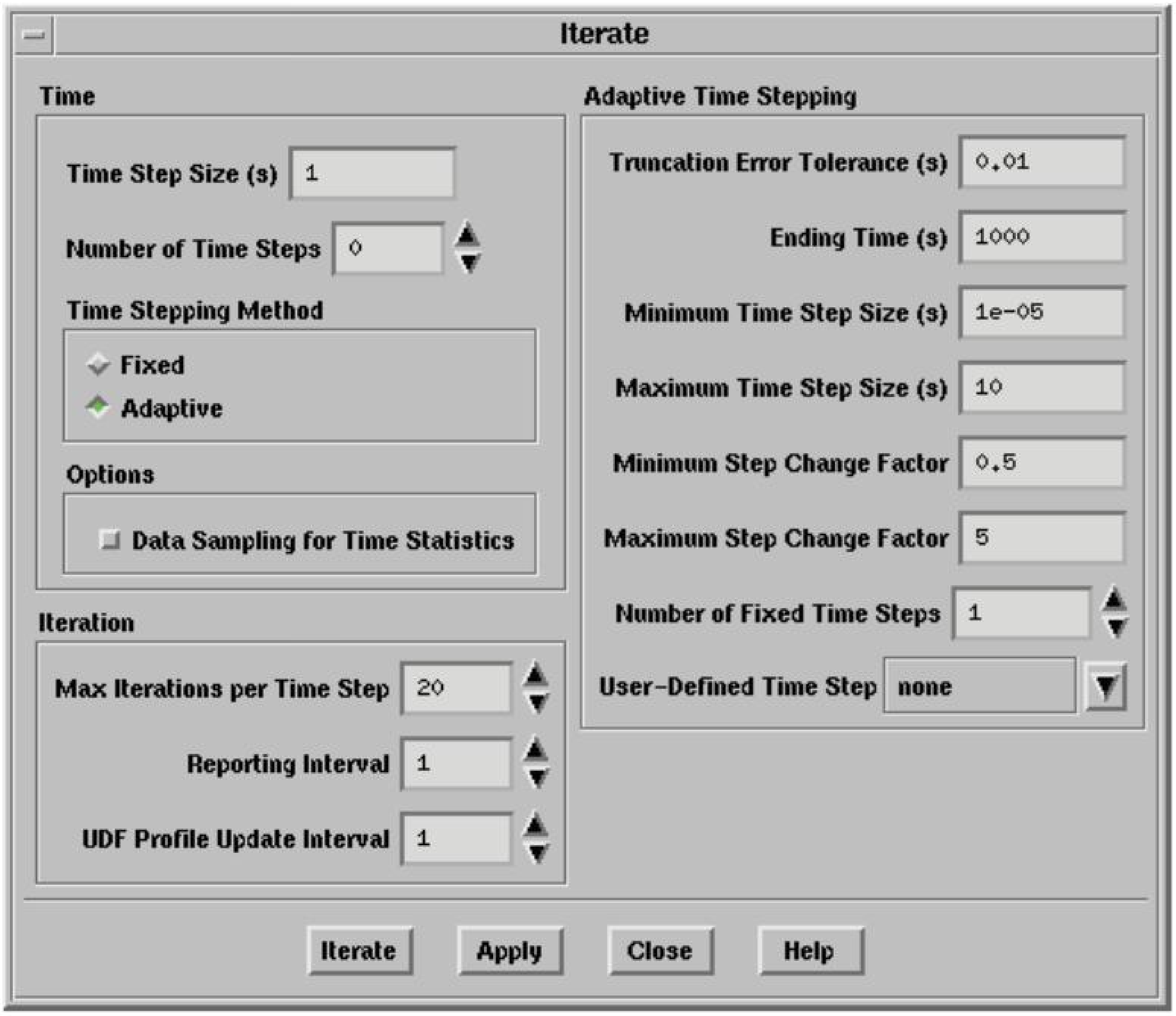


图 6-8 迭代面板

隐式非定常计算公式参数包括每个时间步的 Max Iterations per Time Step（最大迭代次数）和 Time Step Size（时间步长的大小）。虽然从线性稳定性理论分析，全隐式格式是无条件稳定的，但是如前所述，在非线性系统中，时间步长还是要受到一定的限制的。通常在计算开始时，时间步长取得较小，而在后续的计算中则可以逐步增加时间步长。在缺省设置中，时间步长是固定的，如果需要让时间步长随计算进程发生变化，可以选择 Adaptive Time Stepping（适应性时间推进）法中的 Adaptive（适应性步长）选项。

□ 定义计算所需要的 Number of Time Steps（时间步数量），然后点击 Iterate 按钮开始计算。

2）显式非定常计算的步骤如下：

□ 在 Solution Controls（求解过程控制）面板中保留 Solver Parameters（求解器参数）缺省设置：

Solve->Controls->Solution...

如果对 Solver Parameters 进行过修改，可以点击 Default（缺省）按钮恢复缺省设置。

- 输入所需要的 Number of Iterations（迭代步数），点击 Iterate 按钮开始计算：

Solve->Iterate...

（11）保存数据文件和算例文件，以便将来可以继续进行计算：

File->Write->Data...

适应性时间推进

如前所述，时间步长在计算过程中是可以随着截断误差的变化而变化的。在截断误差小于某个给定值时，时间步长将增加。在截断误差大于某个给定值时，时间步长将减小。使用预报校正格式可以计算出截断误差的值，并根据这个值相应调整时间步长。在每个时间步中，预报步可以用耗时较少的显式格式得到一个流场解，这个流场解可以作为校正步的初始条件用于校正步的计算。校正步通常用非线性迭代过程进行计算，校正步和迭代步中得到的两个解的差可以作为截断误差，将这个值与所需要的精度进行对比后，FLUENT 就可以相应增加或减小时间步长。

适应性时间推进的控制参数在 Iterate（迭代）面板中设置，包括：

- Truncation Error Tolerance（截断误差容限），即与截断误差进行对比的判据。增加这个参数的值，则时间步长将增加而计算精度将下降；减小这个值，则时间步长将减小而计算精度将上升。系统设定的缺省值为 0.01。
- Ending Time（结束时间）。因为在适应性时间推进算法中，时间步长是变化的，所以需要事先设定一个结束时间，在累积时间达到结束时间时计算自动结束。
- Minimum/Maximum Time Step Size（最小和最大时间步长），即时间步长的上下限。
- Minimum/Maximum Step Change Factor（最小和最大步长改变因子）是时间步长变化的限制因子，采用这个参数主要是为了限制时间步长发生剧烈变化。如果步长改变因子用 f 表示，并且将其定义为给定截断误差与截断误差容限的比值，则时间步长 Δt_n 用下述方式进行计算：

  在 $1 < f < f_{\max}$ 时，Δt_n 将增加以满足误差容限的要求。

  在 $1 < f_{\max} < f$ 时，Δt_n 将增加，但其最大值为 $f_{\max}\Delta t_{n-1}$。

  在 $f_{\min} < f < 1$ 时，Δt_n 保持不变。

  在 $f < f_{\min} < 1$ 时，Δt_n 将下降。

- Number of Fixed Time Steps（固定时间步的数量），即在时间步长发生变化之前的迭代步数。

6.13 监视计算收敛过程

在计算过程中可以动态监视残差、统计数据、受力值、面积分和体积分等与计算相关的信息，并可以在屏幕上或其他输出设备上输出这些信息。

6.13.1 监视残差

在每个迭代步结束时，都会对计算守恒变量的残差进行计算，计算的结果可以显示在窗口中，并保存在数据文件中，以便随时观察计算的收敛史。从理论上讲，在收敛过程中残差应该无限减小，其极限为 0，但是在实际的计算中，单精度计算的残差最大可以减小 6 个量级，而双精度的残差最大可以减小 12 个量级。

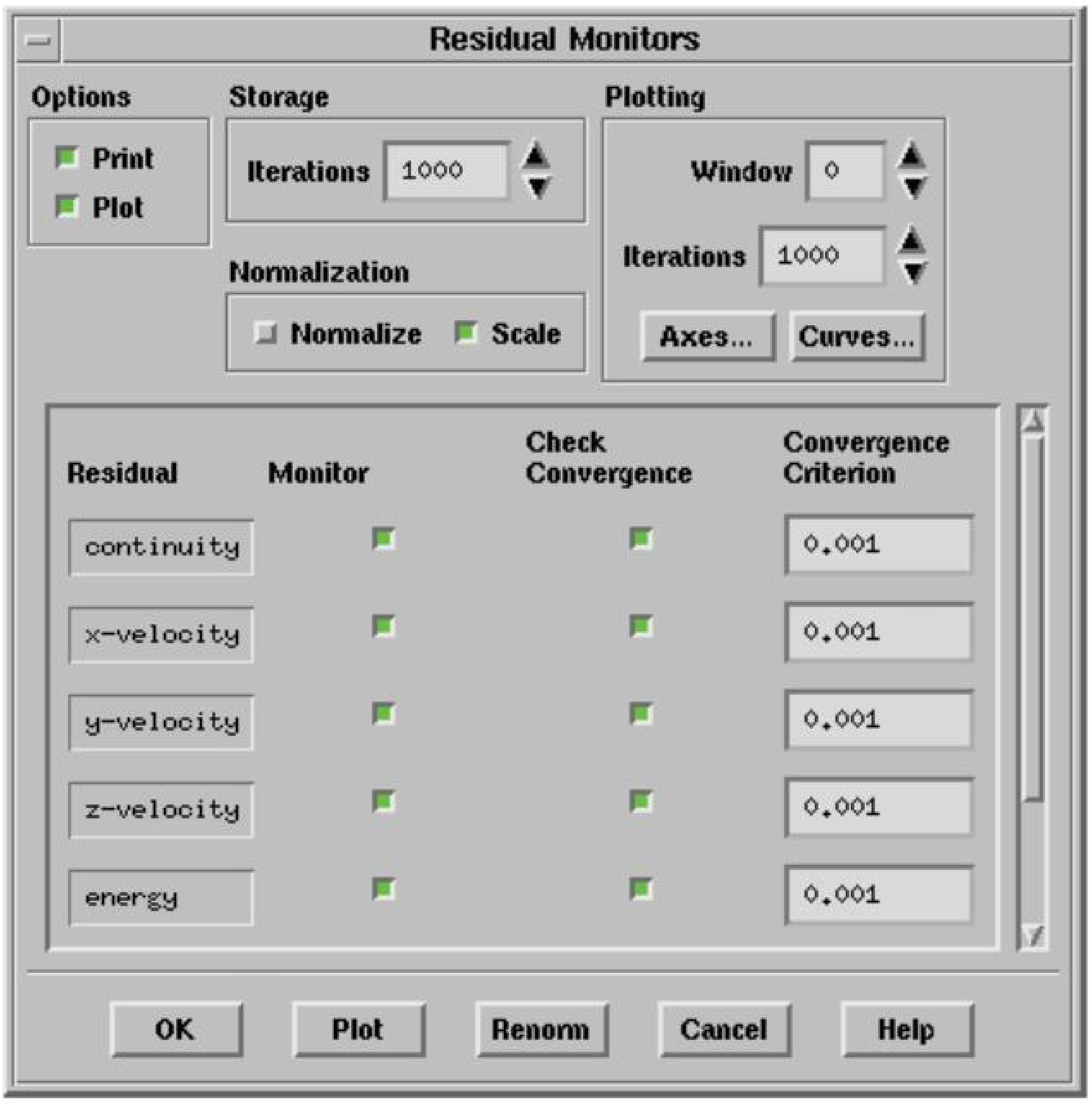


图 6—9 残差监视器面板

分离算法中的残差是守恒方程两端数值的差值在所有网格点上的和，耦合算法中的残差则是各变量的时间变化率在所有网格点上的均方根。在缺省情况下，系统显示的残差是

缩尺残差即残差相对于某个值的比例，其计算方法因变量不同、算法不同而有所不同。而残差的正则化是将残差值除以最大残差值的变换过程。

残差显示设置在 Residual Monitor（残差监视器）面板中进行，如图 6－9 所示，启动残差监视器的菜单操作过程为：

Solve->Monitors->Residual...

或

Plot->Residuals...

在面板的下部可以选择需要监视的变量，并针对各变量设置收敛判据，选择检查该变量是否满足收敛判据等。在面板上方可以选择是否在控制台窗口中以文本方式输出残差的数值（Print 选项），是否绘制残差曲线（Plot 选项），并可以选择保存几个迭代步上的残差值（Iterate 选项），是否对残差进行正则化处理（Normalize 选项），是否进行缩尺处理（Scale 选项），以及残差的窗口显示风格（Plotting 组），包括显示窗口（Window 选项）、显示迭代点数（Iterations）和显示线型、字体等（Axes、Curves）。

6.13.2 监视统计数据

在计算过程中，可以监视周期流动的压强梯度和温度比、非定常流动所用时间、适应性时间推进过程中的时间步长等参数。

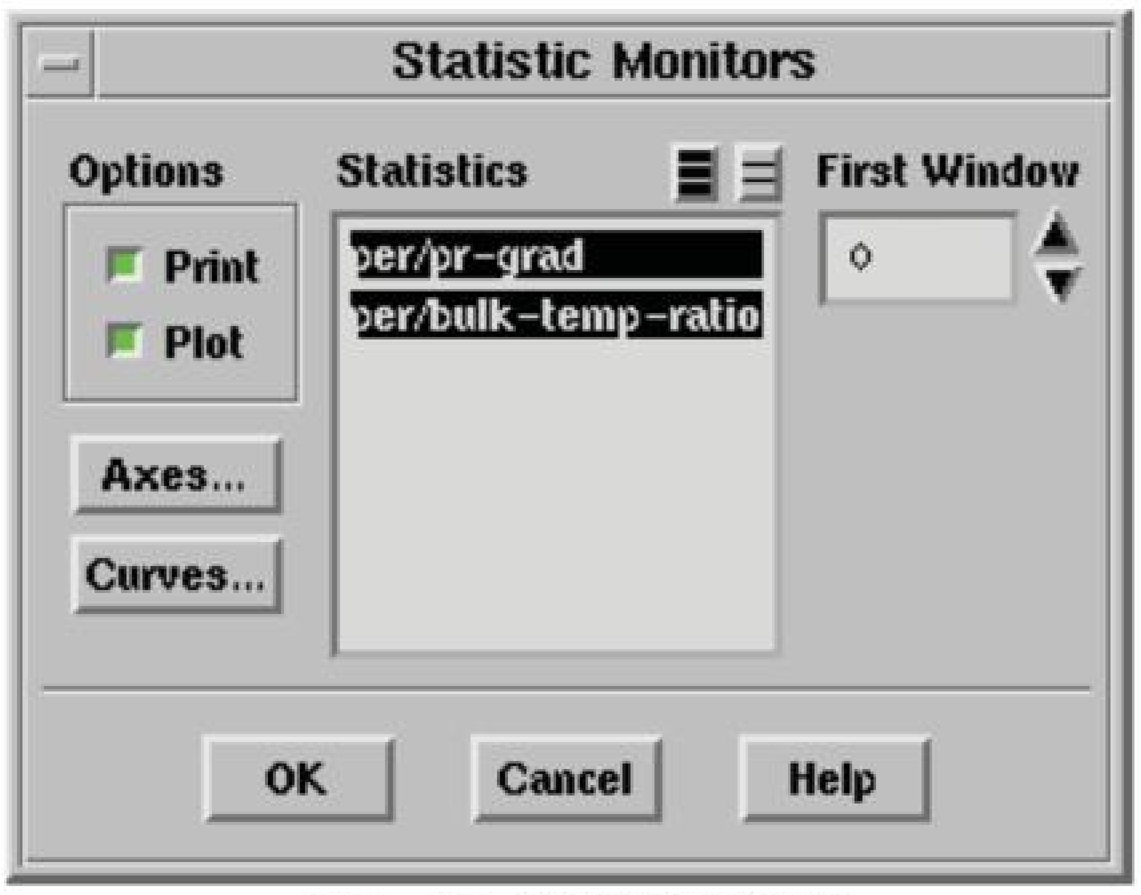


图 6－10 统计数据监视器

统计数据监视器的启动过程如下：

Solve->Monitors->Statistic...

设置监视器的步骤如下：

（1）指定输出类型，即指定是否使用 Print 方式，或 Plot 方式。

（2）在 Statistics 列表中选择需要监视的变量。

（3）如果选择了 Plot（绘图）方式，则可以用 Axes..（坐标轴）面板和 Curves...（曲

线）面板对相关参数进行设置，比如设置显示线型、字体、颜色等等进行设置。

6.13.3 力和力矩监视器

在每次迭代结束后，可以通过计算得到流场中物体所承受的来自流体的力和力矩系数。力和力矩系数也可以通过文本方式（Print 选项）或图形方式（Plot 选项）在屏幕上显示。

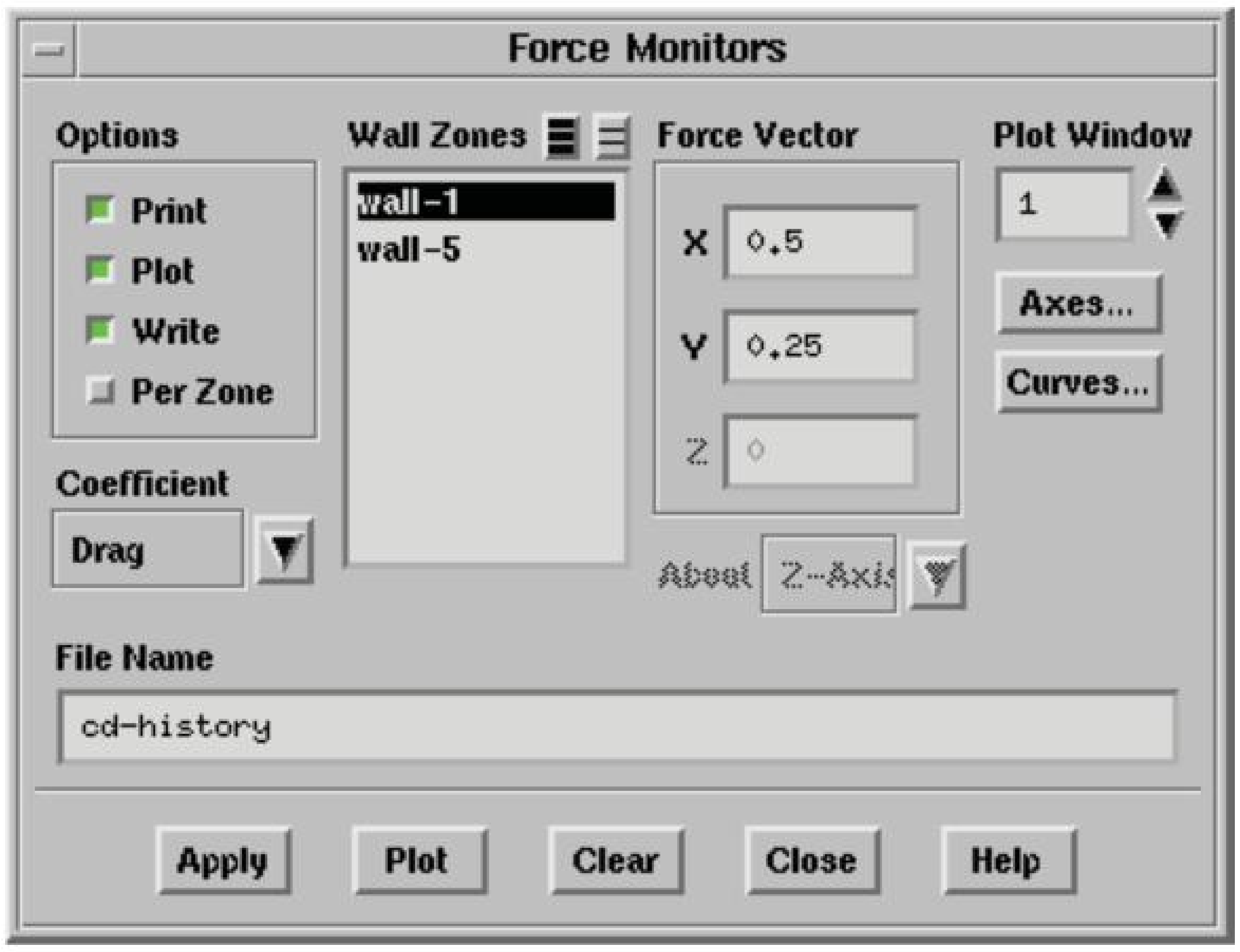


图 6－11 受力监视器

在很多情况下，计算关心的中心问题是物体在流场中受到的力和力矩大小及分布等情况，比如在计算飞机绕流时，最关心的就是飞机行受到的气动力和力矩。在很多情况下，虽然残差仅仅收敛了三个量级，气动力已经收敛。在这种情况下，可以结束计算以节省计算时间，因此对气动力进行监视就很必要。

力和力矩的监视器面板，如图 6－11 所示，通过下列菜单操作被打开：

Solve->Monitors->Force...

相关设置如下：

（1）指定输出类型，即选择以文本形式（Print）输出、图形形式（Plot）输出，还是文件形式（Write）输出。

（2）如果需要对作用在某个壁面上的力和力矩进行监视，可以打开 Per Zone（分区）选项。

（3）在 Coefficient（系数）列表中选择 Drag（阻力）、Lift（升力）或 Moment（力矩），以显示相应的系数。

（4）在 Wall Zones（壁面区）列表中选择壁面名称。

（5）如果选择显示阻力或升力，则在 Force Vector（力矢量）中输入力矢量的 X、Y、Z 分量。如果选择显示力矩，则在 Moment Center（力矩中心）中输入力矩中心的直角坐标值，然后在 About（转动轴方向）列表中选择力矩矢量的方向，即 X-Axis、Y-Axis 或 Z-Axis。

（6）点击 Apply（应用）按钮完成设置。如果需要设置其他参数，则重复上述过程。

6.13.4 监视表面积分

在每次迭代结束后，还可以在某个面上对特定的流场变量进行积分，并以文本、图形和文件形式输出积分结果。比如，在以计算压强为目的的计算中，可以在某个面上监视压强的变化过程。

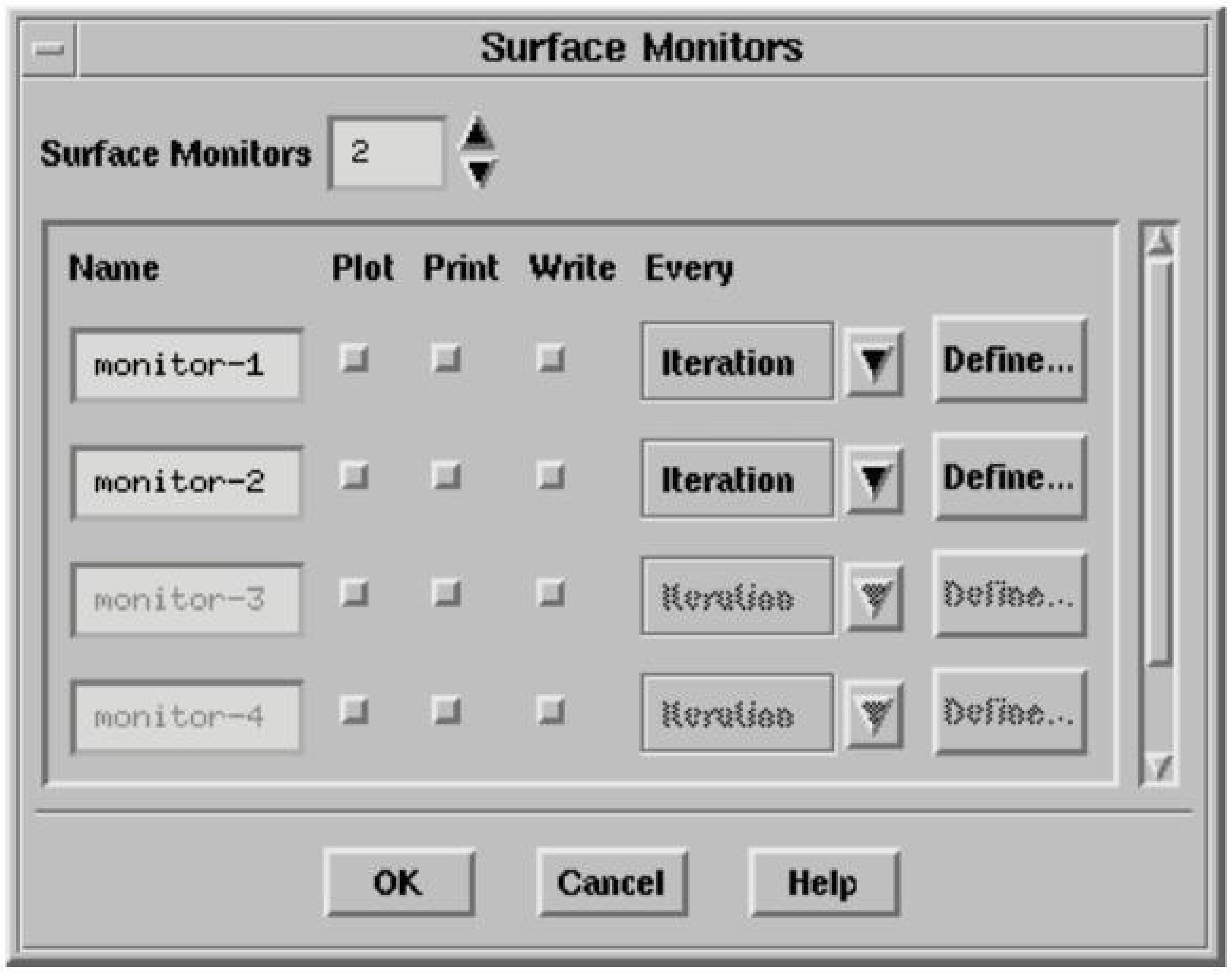


图 6－12 表面监视器

在计算过程中需要监视几个表面上的变量，是在 Surface Monitors（表面监视器）面板中决定的。表面监视器面板的启动操作如下：

Solve->Monitors->Surfaces...

表面监视器的操作过程如下：

（1）在 Surfaces Monitors 旁边的输入栏中填入准备开启的监视器数量，也就是需要监视变量的数量。

（2）在 Name（名称）下面的栏目中加入各监视器的名称。在 Plot、Print、Write 下面选择变量的输出方式，即选择图形方式、文本方式和文件方式。

（3）在 Every 下面选择显示变量的频率，可以选择每个 Iteration（迭代步）、每个 Time Step（时间步）更新一次监视器窗口。

（4）点击 Define（定义）按钮进入 Define Surface Monitors（定义表面监视器）面板，如图 6-13 所示。

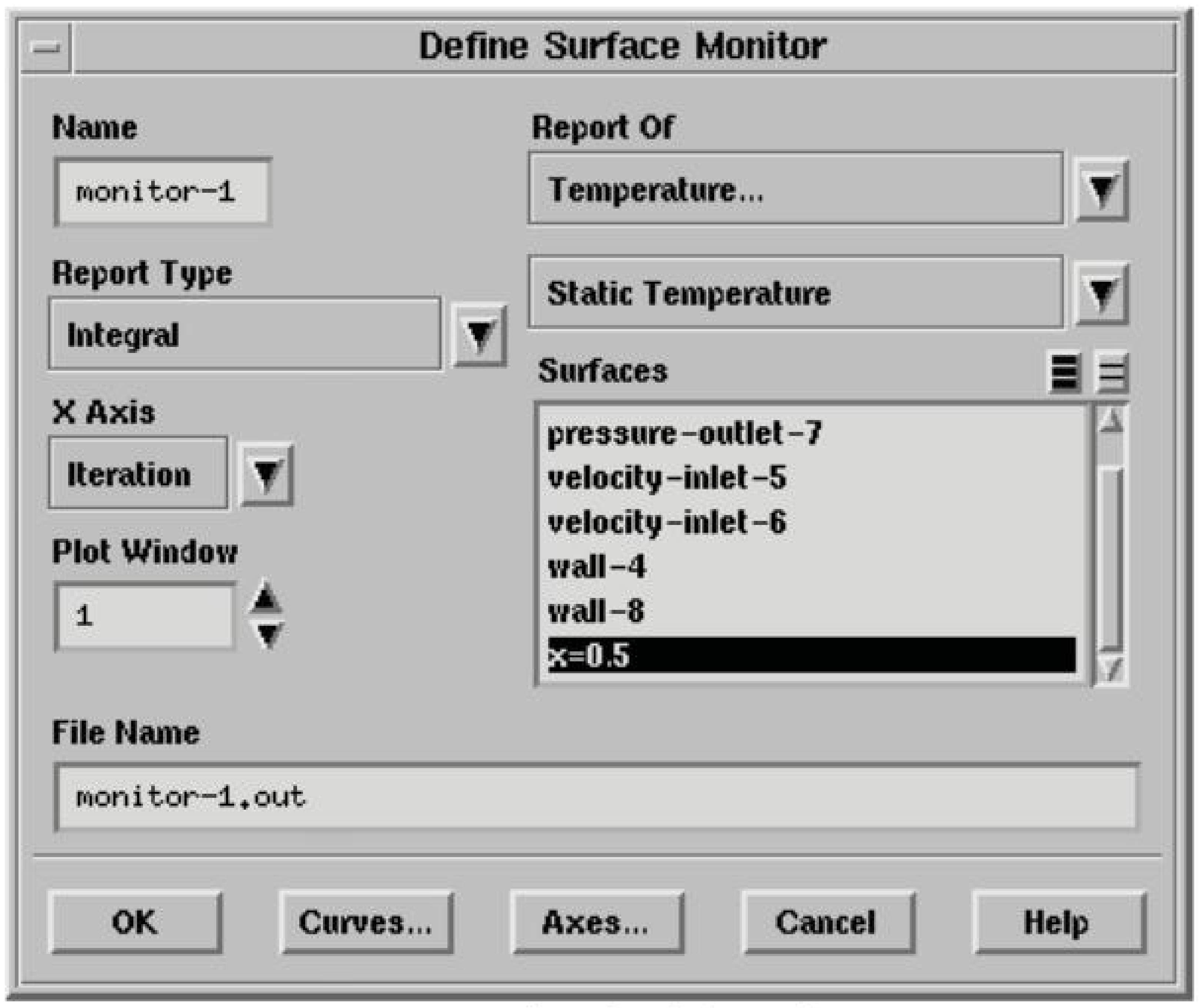


图 6－13 定义表面监视器面板

（6）在 Define Surface Monitors 面板中可以顺序定义监视器名称、报告类型、X 轴变量、显示窗口编号、变量名、积分表面、输出文件名等等，还可以定义图形显示的字体、曲线线型等等。

比如，如果想显示 x=0.5 表面上的静温的积分值，可以先给定监视器的名称，然后在 Report Type 中选择 Integral，在 X Axis 中选择 Iteration，即在显示窗口中 X 轴代表迭代步，在 Report of 中选择 Temperature（温度），在下面的列表中选择 Static Temperature（静温），在 Surfaces 列表中选择 x=0.5，在 File Name 中指定输出文件名，然后点击 OK 即定义完毕。如果对显示效果不满意，可以点击下面的 Curves...按钮和 Axes...按钮对坐标轴及显示曲线等进行详细定义，在大多数情况下没有这个必要。

6.13.5 体积分监视器

与监视面积分类似，在计算过程中还可以监视流场变量的体积分。在体积分监视器中，实际上可以监视的参数还包括流场变量的质量积分、质量平均等等。体积分主要用来监视某个体域内流场变量的变化情况，通过监视变量的体积分可以对求解过程是否收敛做出判断。在使用适应性网格技术的时候，体积分也可以用来判断解是否与网格有关，即在网格

变化的过程中，如果解不随网格变化而变化，则证明计算得到的解与网格无关。

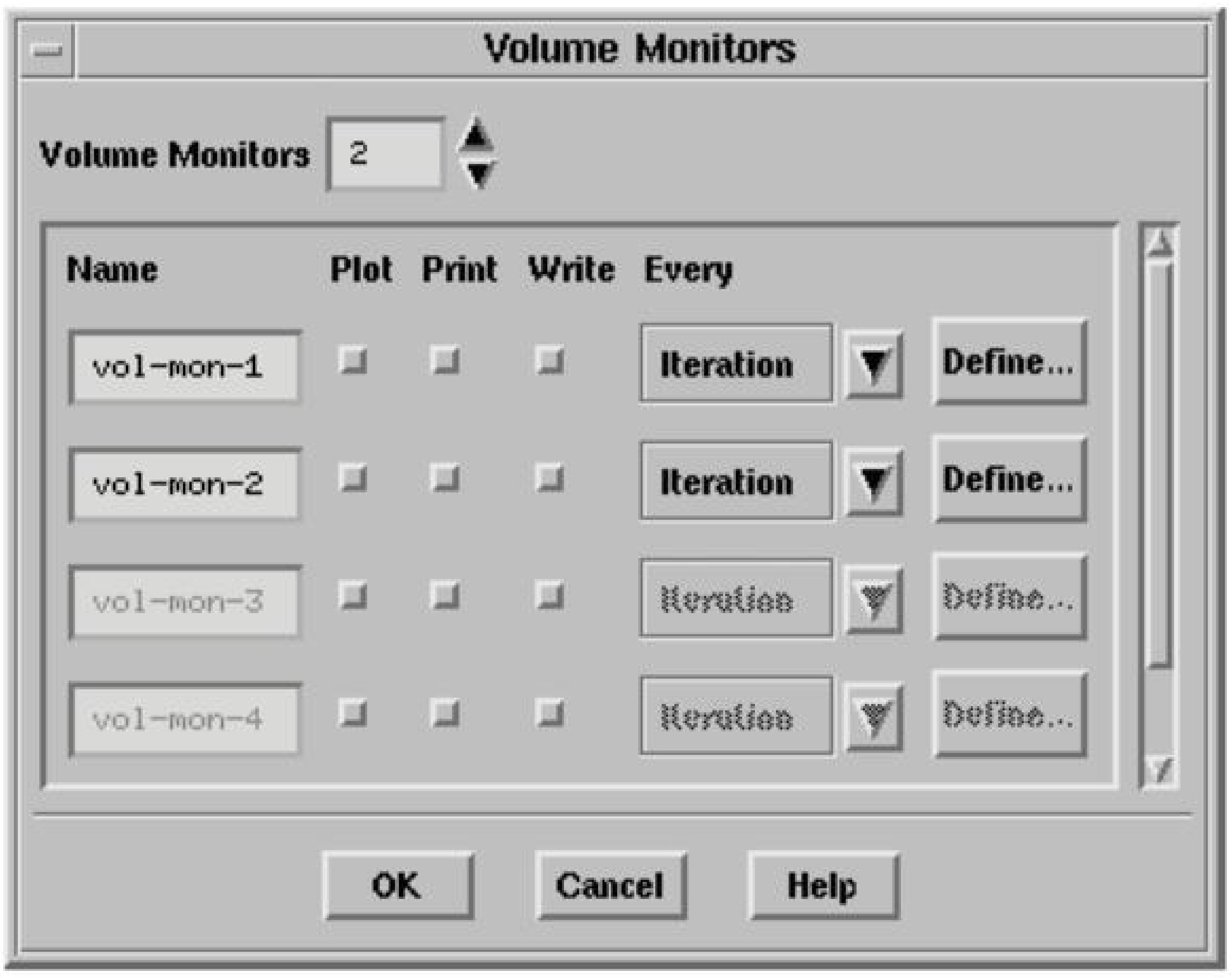


图 6-14 体积分监视器面板

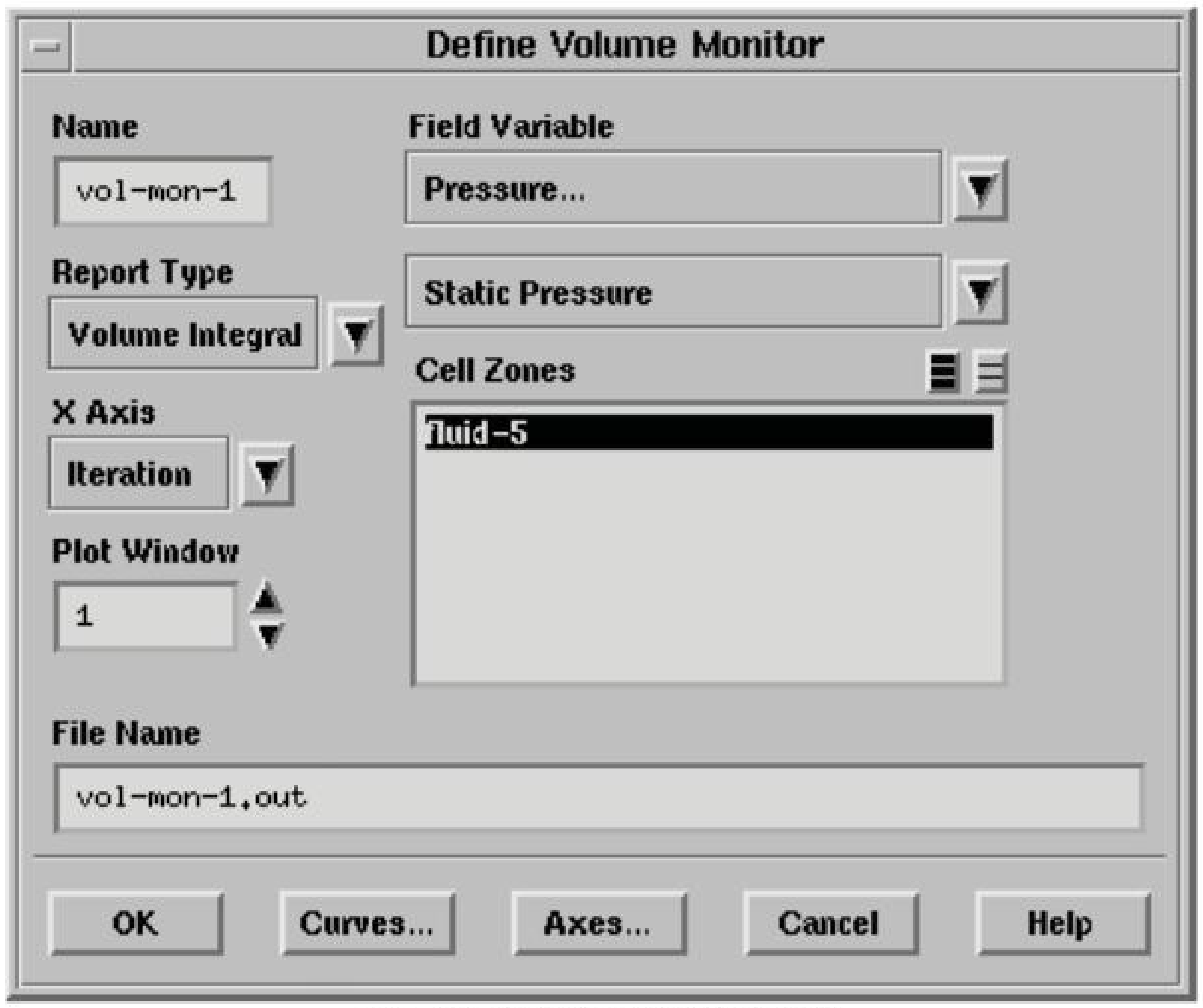


图 6-15 定义体积分监视器面板

体积分监视器在 Volume Monitors（体积分监视器）面板中做总体设置，在 Define Volume

Monitors（定义体积分）面板中做详细设置。体积分监视器的启动过程为：

Solve->Monitors->Volume...

体积分监视器面板与面积分监视器面板是十分相似的，其设置过程也类似，即可以在这里设定窗口名称、窗口数量、显示方式等选项，并通过点击 Define（定义）按钮进入 Define Volume Monitors（定义体积分监视器）面板，如图 6-15 所示。在这个面板中可以指定准备做积分的体域、流场变量、报告类型等等，与面积分定义过程相类似，这里不再重复。

6.14 用动画显示解

在 FLUENT 中可以用动画形式显示变量、网格、曲线等的演化过程，生成的动画文件可以用 MPEG 格式保存，并在各种媒体播放器上重放。

6.14.1 定义动画演示序列

动画演示与前面介绍的监视器显示在很多方面是类似的，主要区别是动画是动态过程，而监视器显示的内容虽然也是变化的，但是不如动画生动。

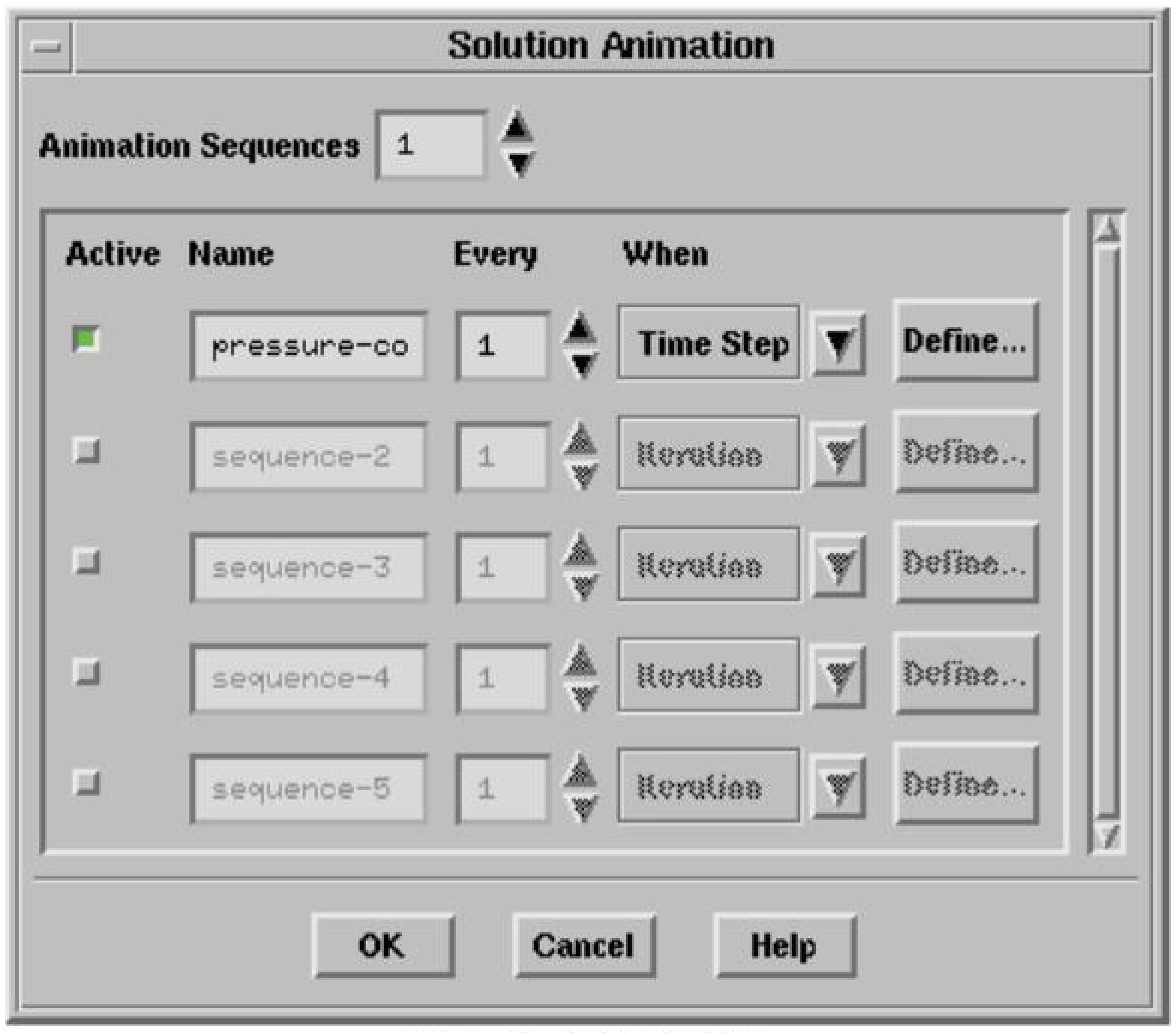


图 6-16 求解动画面板

与监视器的设置过程类似，动画的设置也是在两个面板中完成的。首先启动 Solution

Animation（求解动画）面板：

Solve->Animate->Define...

在这个面板中，可以指定创建动画的数量（Animation Sequences），然后打开 Active（激活）按钮，指定每个动画的名称及采样间隔，最后点击 Define（定义）按钮进入 Animation Sequence（动画序列）面板。

在动画序列面板中，主要设定动画的 Storage Type（保存类型）和 Display Type（显示类型）。

FLUENT 中保存动画的方式包括内存（In Memory）、图元（Metafile）和位图（PPM Image）三种方式。如果生成的动画不大，可以选择保存在内存中。如果需要将动画文件保存在硬盘上，可以选择图元或位图方式。其中图元格式的动画文件占用硬盘空间比较大，如果硬盘空间不大，最好选择位图方式。

显示类型包括网格（Grid）、等值线（Contours）、矢量（Vector）、XY 曲线（XY Plot）和监视器（Monitor）五种形式，用户可以根据所显示的变量的不同而有所变化，比如在显示适应性网格变化过程的动画中可以选择 Grid 形式；在显示压强场时可以选择 Contours 形式；在显示速度场时可以选择 Vector 形式；在显示残差变化时可以选用 XY Plot 形式等等。

在确定了显示类型后，可以点击 Properties...按钮进行详细设置，这里不再细述。

6.14.2 播放动画

Playback（动画播放）面板的启动方式为：

Solve->Animate->Playback...

如图 6-17 所示，动画播放面板与常用的媒体播放软件十分相象。在动画列表中选择一个动画文件，然后点击播放键就可以观看动画。在这个面板上可以设置播放的速度和播放的动画范围，也可以一帧一帧地进行播放。

比如计算进行了 5000 步，动画设置中设定的是每个时间步绘制一个动画帧，则最后的动画文件是由 5000 帧画面组成的。这样一个动画播放起来可能很费时，因此在第一次观看的时候可以拖动速度滑块，让动画以快速形式播放。如果在观看的过程中对某一段动画感兴趣的话，可以选择动画的开始（Start Frame）帧和结束帧（End Frame），缩小播放范围仔细观看。如果在观看过程中发现某一帧中的某个图像细节需要仔细研究的话，可以选择逐帧播放的形式，一帧一帧地仔细研究图像细节的变化。

在动画播放的过程中，可以用鼠标拖动的方式对画面进行平移、旋转、缩放等操作。如果对动画不满意，可以点击 Delete 键删除动画文件。还可以在 Play Mode（播放模式）列表中选择 Play Once（播放一次）、Auto Repeat（自动重播）、自动回放（Auto Reverse）等等。

在 Write/Record Format（写/记录格式）中，可以选择动画文件的保存格式。如果前面选择了用内存方式保存文件的话，可以在这里将内存中的动画文件转为硬盘文件。

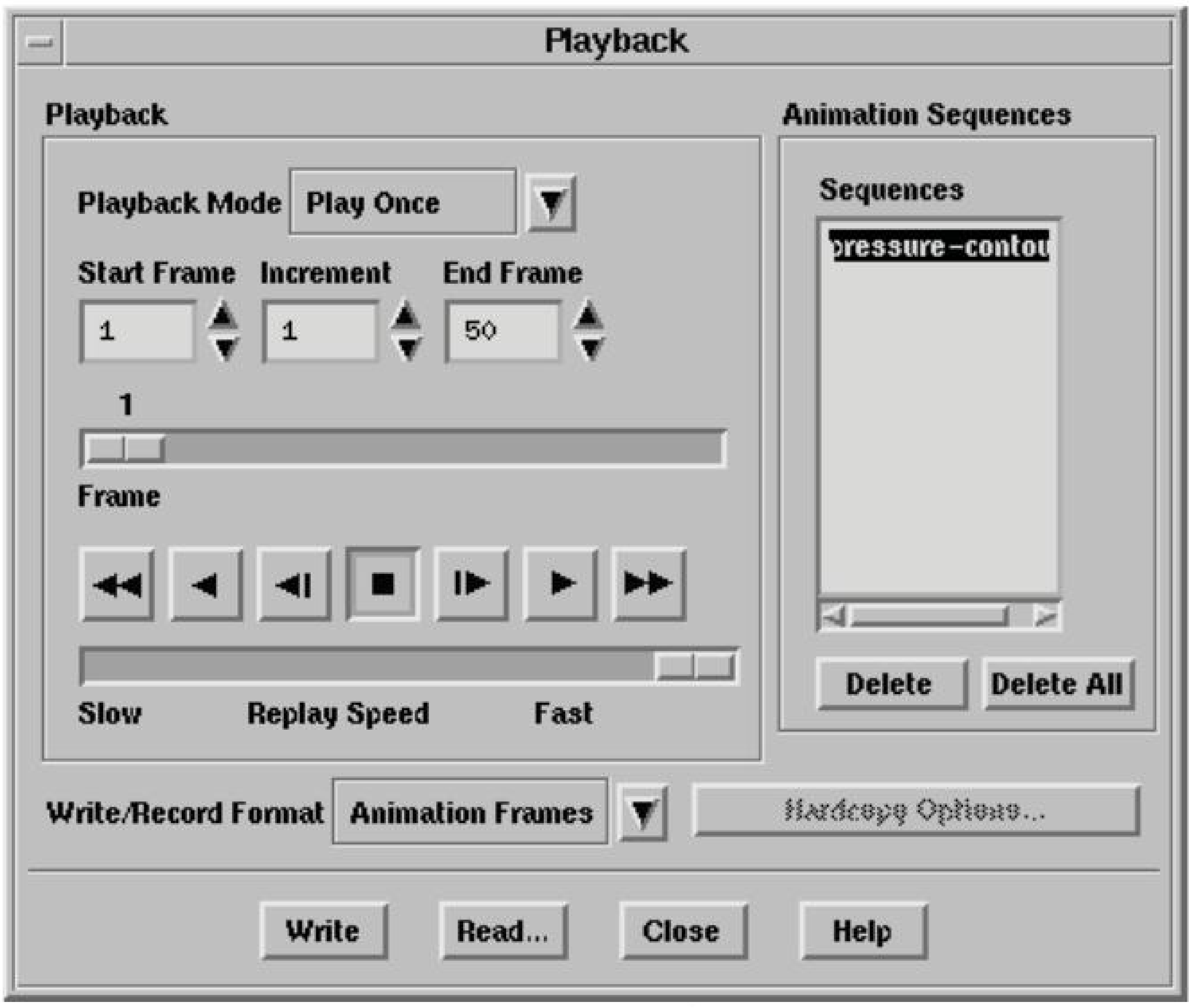


图 6-17 Playback（动画播放）面板

如果动画文件已经存在于硬盘上，则可以用播放面板直接将它读入，而不必读入相应的算例文件和数据文件。

6.15 在计算过程中执行命令

在 FLUENT 的计算过程中，每隔一段时间间隔可以执行文本命令或宏命令。这项功能在 Execute Commands（执行命令）面板中进行设置，这个面板的启动方法为：

Solve->Execute Commands...

如图 6-18 所示，在这个面板中可以指定需要执行的命令数（Defined Commands），然后激活命令栏，定义命令执行的时间间隔，再写入文本命令或宏命令的名称即可。

宏命令是一个可以自动执行的命令序列。如果需要在一定的间隔后进行一系列的命令操作，则可以先录制宏命令，然后在命令栏中填入宏命令的名称。点击面板下方的 Define Macros...（定义宏命令）按钮可以进入 Define Macro（宏命令定义）面板，如图 6-19 所示。

在宏命令定义面板中，首先设定宏命令的名称，然后点击 OK 按钮开始宏命令录制过程。在接下来的时间里，依次进行相关操作，比如让网格进行适应性调整。录制完毕后，在执行命令面板中点击 End Macro（结束宏命令）按钮结束录制过程。在后面的计算中就可以引用这个宏命令完成相关的操作。

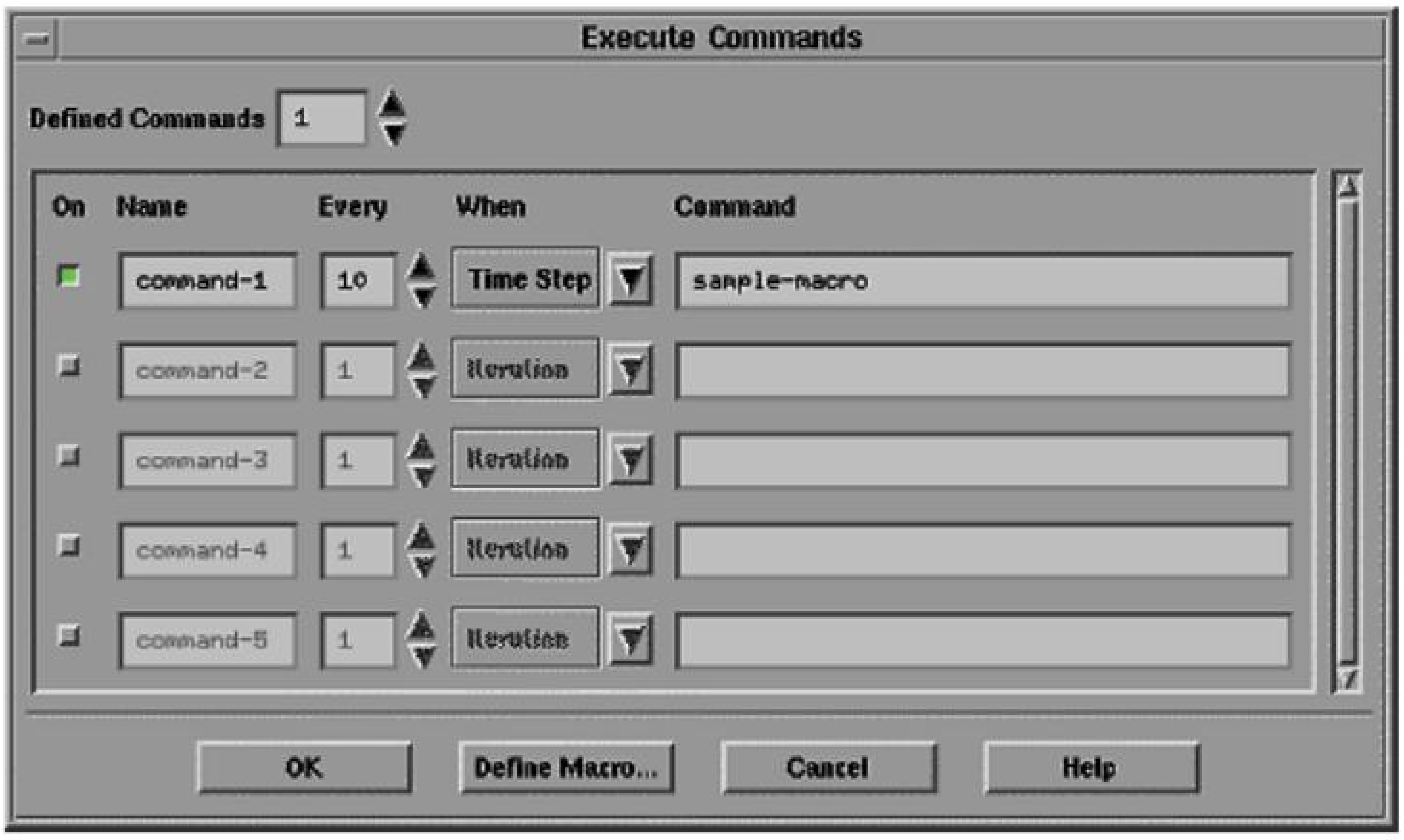


图 6-18 执行命令面板



图 6-19 宏命令定义面板

如果计算过程中需要保存算例文件和数据文件，则应该用 Autosave Case/Data（自动保存算例和数据文件）面板，其菜单操作顺序为：

File -> Write -> Autosave...

6.16 收敛性和稳定性

6.16.1 判断收敛

判断收敛性的常规方法是观察残差的变化——如果残差曲线下降则意味着计算收敛，反之则意味着发散。除此之外，还可以观察流场变量的变动情况，如果重要的流场变量在经过一段时间的计算不再变化，则意味着计算已经收敛，可以结束计算。

影响收敛性的因素很多，包括亚松弛因子、初始流场、物理模型等都会对计算是否收敛形成直接影响。正因为如此，所以很难给出一个统一的收敛判据。在大多数情况下，残差下降 3 个数量级就可以认为计算已经收敛了，但是如果初始流场非常靠近最终解，同时残差计算以初始场为基准进行比例计算，则残差下降的幅度可能不会达到 3 个数量级。与此相反，如果初始流场设置的非常粗糙，则在达到收敛时残差的下降幅度可能又远远超过 3 个数量级。因此，在用残差下降幅度判断收敛之外，还可以用残差在计算过程中的行为进行收敛性判断，即残差持续下降时间或迭代步数达到一定的值，就可以认为计算已经收敛了。另外，因为在定常计算中，“收敛”的真实含义就是流场变量的变化幅度已经足够小，以致我们可以认为流场不再变化，所以也可以直接用流场变量的变化幅度作为计算是否收敛的判据。

6.16.2 加速收敛

在计算发散时最容易想到的解决办法是减小亚松弛因子的值，在很多情况下，这个办法是有效的，因为计算发散的最常见原因是由方程的非线性引起的，而亚松弛因子是用线性稳定性分析得到的，因此减小亚松弛因子可以让计算回到稳定域中。

计算发散还可能由初始流场的设置引起，即如果初始流场给的过分粗糙，也有可能引起计算的发散。在这种情况下，解决问题的办法当然应该从改善初始流场入手。如果计算中涉及的物理模型非常复杂，可以考虑先计算简化流场，再以简化流场为初始流场继续进行计算。比如，在多相流计算中，可以先计算连续相流场，然后再加入弥散相进行计算。在带有化学反应的流场计算中，可以先计算混合物流场，或冻结流流场，再计算带化学反应模型的流场，等等。

6.16.3 修改多步时间推进参数

控制多步时间推进格式最常用的参数是库朗（Courant）数，除此之外，还可以通过残差光顺和改变多步格式的方法增加计算稳定性或增加时间步长。

用残差光顺的方法增加库朗数

在 Solution Controls（求解过程控制）面板中，残差光顺的迭代值在缺省设置中被设定为 0，即在缺省设置中没有使用残差光顺技术。如果将 Iterations（迭代计数器）增加为 1 或更大的数，则可以进一步设置 Smoothing Factor（光顺因子）。将光顺因子设定为 0.5 可以将库朗数增加为原数值的两倍。

改变多步格式

首先启动 Multi-Stage Parameters（多步格式参数）面板：

Solve->Controls->Multi-Stage...

在这个面板中，可以增加多步格式的步数，重新设置每一步的系数，并决定是否每步更新耗散项和粘性应力的值。



图 6-20 多步格式参数面板

在缺省设置中，FLUENT 使用 5 步格式，每步的系数分别为 0.25、0.166666、0.375、0.5 和 1.0。在对多步格式非常熟悉的情况下可以增加多步格式的步数，同时修改每步的系数。修改系数的一般要求是：

（1）系数为介于 0 和 1 之间的实数。

（2）最后一步的系数必须为 1。

在多步格式参数面板上还可以决定是否在每一步都计算耗散项和粘性项。选中耗散项

意味着在相应的计算步中需要计算人工耗散，如果不选，则在相应的计算步中人工耗散项保持与前一步相同的值，即人工耗散项被“冻结”。粘性项的情况与此类似，在被选中时计算粘性应力，不选时，粘性应力被冻结。需要注意的是，在第一步计算中应该计算粘性应力，随后的计算中如果继续计算粘性应力可以增加计算的鲁棒性，但同时需要消耗更多的 CPU 时间。

在希望恢复缺省设置时，可以点击 Default（缺省）按钮，此时所有参数值恢复缺省值，同时 Default（缺省）按钮变为 Reset（重置）按钮。如果希望将缺省值恢复为设定的值，可以点击 Reset 按钮，则所有参数值可以重新恢复人为修改过的参数值。

第七章 FLUENT 的物理模型

FLUENT6.1 中采用的物理模型包括基本流动模型、湍流模型、动网格模型、化学反应模型、燃烧模型、PDF 模型、弥散相模型、多相流模型、热交换模型、气动噪声模型和固化与熔化模型等等，可以模拟包括可压、不可压、定常、非定常等流动范围的问题，涉及的工业应用领域包括加工设备中的层流非牛顿流体问题、旋转机械和风扇问题、空气动力学内外流问题、多相流问题、多孔介质问题、气动噪声问题等等。限于本书篇幅，本章仅介绍与软件使用相关的内容，更深入的理论知识，请读者参阅有关专业方向的参考资料。

7.1 基本流动模型

FLUENT 在所有问题的求解过程中都会求解质量和动量的守恒方程。在涉及到可压缩性和热交换的问题时则求解能量方程。如果计算问题中包含组分输运过程，比如在化学反应流，或混合物流动的计算中，则增加对组分浓度方程的求解。在包含湍流模型的问题中，还会增加对湍流动能等湍流变量的守恒方程的求解。

7.1.1 周期性流动

周期性流动即流动中包含周期性特征的流动。周期性流动特征则是在一定的时间、空间间隔上流场变量或其导数重复出现的现象。当然从严格的数学意义上说，周期性流动中流场变量或其导数值可能只是在一定的时间、空间间隔上出现近似解，而不一定是绝对相等的解。如果严格按照数学定义来定义周期性，显然会增加软件设计的难度，因此 FLUENT 中所说的周期性是绝对相等的变量或其增量在一定时间、空间间隔上重复出现的现象。

如边界条件一章中所述，周期性边界条件包括旋转周期性条件和平移边界条件，即沿圆周方向呈周期性流动特点，或沿直线方向呈周期性流动特点。周期性流动的另一个分类方法是看流动过程中是否存在压力降，即有压力降的周期性流动和无压力降的周期性流动。本章讨论的流向周期性流动（streamwise periodic flows），指的是沿流动方向存在周期性特征，同时不存在压力降的情况。在周期性流动中，相隔固定空间距离的速度分量和压强的值是相等的。

在周期性计算中，流动必须满足的条件包括：

（1）流动为不可压流。

（2）流场在流动方向上的几何特征具有周期性特征。

（3）如果采用耦合求解器，只能给定压强增量。如果采用分离求解器，则可以同时给

定压强增量和质量流量。

（4）在流场入口和出口上没有质量的净增量。

（5）弥散相模型、多相流模型和反应流模型不能出现在计算中。

在采用分离求解器时，周期性流动计算的设置面板为周期性条件（Periodicity Conditions）面板：

Define->Periodic Conditions...

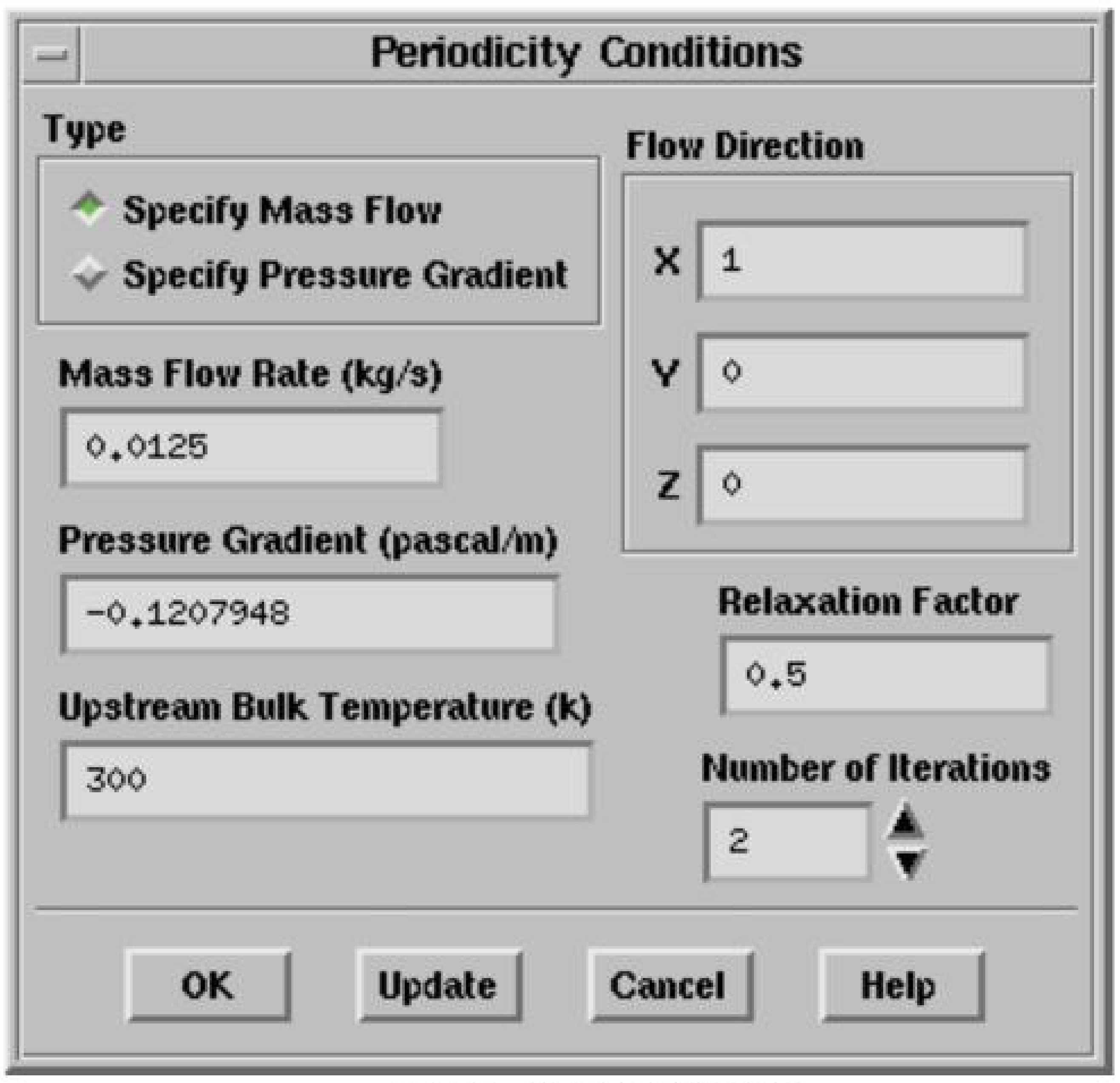


图 7-1 周期性条件面板

设定过程中首先选择采用质量流（Specify Mass Flow）或压强梯度（Specify Pressure Gradient）定义方式。然后相应地在下面栏目中输入质量流或压强梯度的值，并定义流场温度和方向。假设采用质量流方式，则压强梯度要通过计算获得，此时定义松弛因子（Relaxation Factor）和迭代次数（Number of Iteratons），就可以通过系统计算出压强梯度的值。

在采用耦合求解器时，应该在周期性（Periodic）面板中进行设置：

Define->Boundary Conditions...

如图 7-2 所示，在周期性面板中在选定的区域上设定周期性计算的类型，即平移型还是旋转型，同时给出周期性压强增量的值，点击 OK 按钮即可。

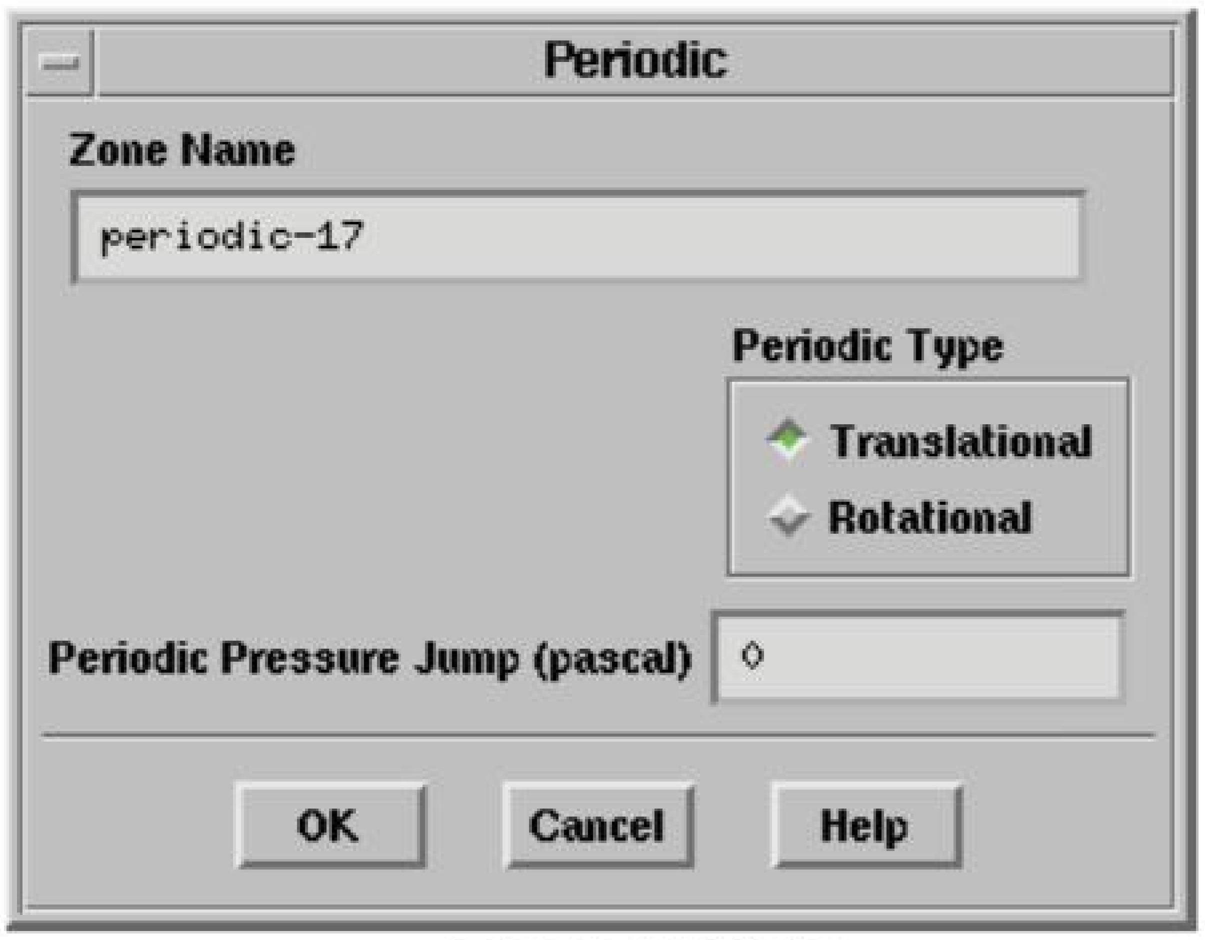


图 7-2 周期性面板

7.1.2 旋转流动

旋转流动是工程应用方面常见的问题类型之一，其流场特征就是流场是由几个大的集中涡组成。在旋转流动中，$k-\varepsilon$ 湍流标准模型失效，因此需要采用 RNG $k-\varepsilon$ 模型、realizable $k-\varepsilon$ 模型做湍流计算。在旋转特别强烈时，建议采用 RSM 模型计算湍流。旋转流动对网格精度要求较高，因此精细的网格对于提高旋转流动的计算精度是必不可少的。在计算轴对称问题时需要注意的是，FLUENT 中规定只能用 X 轴作对称轴，同时网格必须位于 X 轴的上方，即 Y 坐标大于 0 的区域。

轴对称计算需要在求解器（Solver）面板和边界条件（Boundary Conditions）面板中进行设置，即在边界条件面板中的 Space 选项中打开 Axisymmetric Swirl，同时在入口或壁面边界条件中定义旋转边界的转动速度。

求解轴对称旋转流动与求解其他类型的复杂流动问题一样，都可以采取从简单到复杂的计算策略，即为了保证计算的稳定性，可以先计算旋转不太强烈的流场，再逐渐增加旋转强度的方式，逐渐增加流场复杂性，以便保证最后得到高精度的结果。

在后处理过程中可以报告旋转速度（Swirl Velocity）和旋转壁面的剪切应力（Swirl-Wall Shear Stress）。

7.1.3 可压缩流

可压缩流计算通常用于气流流动计算，液体流动在绝大多数情况下被认为是不可压的。衡量流体可压缩性的指标是马赫数。在马赫数小于 0.3 时，气体通常被认为是不可压流体。在马赫数大于 0.3 时，气体的压缩性影响逐渐增强，此时计算中必须考虑压缩性影响。

可压缩流计算中各项参数的设置过程与其他流动计算大同小异，主要区别是在计算中

引入了粘性加热效应，即需要在定义粘性影响和湍流模型的 Viscous 面板中，打开 Viscous Heating 选项，同时需要在物质属性设置中打开理想气体选项，即在 Material 面板的 Density 旁边打开 Ideal-gas 选项。

在计算域中的压强随时间缓慢增长时，可以在操作条件中打开浮动操作压强（floating operating pressure）选项以减小计算中的舍入误差。需要注意的是，这个选项在使用压强入口、压强出口、排气风扇、进气风扇、吸入式风扇、通风出口或压强远场条件等边界条件时不能使用。

可压缩流计算中需要设定的边界条件如下：

（1）在流场入口：

- 压强入口条件：给定入口总温、总压，在超音速入口时，还需设定静压。
- 质量流入口：给定入口质量流和总温。

（2）在流场出口：

- 压强出口条件：给定出口静压。

在计算结束后，可以报告的项目包括总温、总压和马赫数等参数。

7.1.4 无粘流

无粘流假设适用于粘性对流场影响可以忽略的计算中，比如飞行器的气动计算中，如果仅需要计算升力的话，则可以考虑适用无粘流假设计算。无粘流计算的另一个用途是为复杂流动计算提供初始流场。

无粘流计算的设置是在 Viscous 面板中进行的，在面板中选择 Inviscid 选项，并在边界条件和物质属性中进行相应的设置就可以进行计算了。

在后处理过程中，可以报告物体的受力：

Report->Forces...

7.2 湍流模型

因为湍流现象是高度复杂的，所以至今还没有一种方法能够全面、准确地对所有流动问题中的湍流现象进行模拟。在涉及湍流的计算中，都要对湍流模型的模拟能力以及计算所需系统资源进行综合考虑后，再选择合适的湍流模型进行模拟。FLUENT 中采用的湍流模拟方法包括 Spalart-Allmaras 模型、standard（标准）$k-\varepsilon$ 模型、RNG（重整化群）$k-\varepsilon$ 模型、Realizable（现实）$k-\varepsilon$ 模型、v^2-f 模型、RSM（Reynolds Stress Model，雷诺应力模型）模型和 LES（Large Eddy Simulation，大涡模拟）方法。本小节将介绍各种湍流计算方法的特点与使用限制，为在计算中正确选择湍流计算方法提供简单的背景知识。更多的与湍流相关的内容还要参考相关领域的文献资料。

7.2.1 雷诺平均与大涡模拟的对比

因为直接求解 NS 方程非常困难，所以通常用两种办法对湍流进行模拟，即对 NS 方程进行雷诺平均和滤波处理。这两种方法都会增加新的未知量，因此需要相应增加控制方程的数量，以便保证未知数的数量与方程数量相同，达到封闭方程组的目的。

雷诺平均 NS 方程是流场平均变量的控制方程，其相关的模拟理论被称为湍流模式理论。湍流模式理论假定湍流中的流场变量由一个时均量和一个脉动量组成，以此观点处理 NS 方程可以得出雷诺平均 NS 方程（简称 RNS 方程）。在引入 Boussinesq 假设，即认为湍流雷诺应力与应变成正比之后，湍流计算就归结为对雷诺应力与应变之间的比例系数（即湍流粘性系数）的计算。根据计算中使用的变量数目和方程数目的不同，湍流模式理论中所包含的湍流模型又被分为二方程模型、一方程模型和零方程模型（代数模型）等大类。FLUENT 中使用的三种 $k-\varepsilon$ 模型、Spalart-Allmaras 模型、$k-\omega$ 模型及雷诺应力模型（RSM）等都属于湍流模式理论。

大涡模拟（LES）方法是通过滤波处理计算湍流的，其主要思想是大涡结构（又称拟序结构）受流场影响较大，小涡则可以认为是各向同性的，因而可以将大涡计算与小涡计算分开处理，并用统一的模型计算小涡。在这个思想下，大涡模拟通过滤波处理，首先将小于某个尺度的旋涡从流场中过滤掉，只计算大涡，然后通过求解附加方程得到小涡的解。过滤尺度一般就取为网格尺度。显然这种方法比直接求解 NS 方程的 DNS 方程效率更高，消耗系统资源更少，但却比湍流模式方法更精确。尤其应该注意的是，湍流模式理论无法准确模拟大涡结构，因此在需要模拟大涡结构时，只能采用 LES 方法[1]。

尽管大涡模拟理论比湍流模式理论更精确，但是因为大涡模拟需要使用高精度的网格，对计算机资源的要求比较高，所以还不能在工程计算中被广泛使用。在绝大多数情况下，湍流计算还要采用湍流模式理论，大涡模拟则可以在计算资源足够丰富的时候尝试使用。

7.2.2 Spalart-Allmaras 模型

Spalart-Allmaras 模型是一方程模型里面最成功的一个模型，最早被用于有壁面限制情况的流动计算中，特别在存在逆压梯度的流动区域内，对边界层的计算效果较好，因此经常被用于流动分离区附近的计算，后来在涡轮机械的计算中也得到广泛应用。

最早的 Spalart-Allmaras 模型是用于低雷诺数流计算的，特别是在需要准确计算边界层粘性影响的问题中效果较好。FLUENT 对 Spalart-Allmaras 进行了改进，主要改进是可以在网格精度不高时使用壁面函数。在湍流对流场影响不大，同时网格较粗糙时，可以选用这个模型。

Spalart-Allmaras 模型是一种新出现的湍流模型，在工程应用问题中还没有出现多少成功的算例。如同其他一方程模型一样，Spalart-Allmaras 模型的稳定性也比较差，在计算中

[1] FLUENT 中尚未提供 DNS 计算选项。

采用 Spalart-Allmaras 模型时需要注意这个特点。

7.2.3 标准 $k-\varepsilon$ 模型

标准 $k-\varepsilon$ 模型由 Launder 和 Spalding 提出，模型本身具有的稳定性、经济性和比较高的计算精度使之成为湍流模型中应用范围最广、也最为人熟知的一个模型。标准 $k-\varepsilon$ 模型通过求解湍流动能（k）方程和湍流耗散率（ε）方程，得到 k 和 ε 的解，然后再用 k 和 ε 的值计算湍流粘度，最终通过 Boussinesq 假设得到雷诺应力的解。

虽然得到了最广泛的使用，但因为标准 $k-\varepsilon$ 模型假定湍流为各向同性的均匀湍流，所以在旋流（swirl flow）等非均匀湍流问题的计算中存在较大误差，因此后来又发展出很多 $k-\varepsilon$ 模型的改进模型，其中包括 RNG（重整化群）$k-\varepsilon$ 模型和 Realizable（现实）$k-\varepsilon$ 模型等衍生模型。

7.2.4 RNG $k-\varepsilon$ 模型

RNG $k-\varepsilon$ 模型在形式上类似于标准 $k-\varepsilon$ 模型，但是在计算功能上强于标准 $k-\varepsilon$ 模型，其改进措施主要有：

（1）在 ε 方程中增加了一个附加项，使得在计算速度梯度较大的流场时精度更高。

（2）模型中考虑了旋转效应，因此对强旋转流动计算精度也得到提高。

（3）模型中包含了计算湍流 Prandtl 数的解析公式，而不象标准 $k-\varepsilon$ 模型仅用用户定义的常数。

（4）标准 $k-\varepsilon$ 模型是一个高雷诺数模型，而重整化群 $k-\varepsilon$ 模型在对近壁区进行适当处理后可以计算低雷诺数效应。

7.2.5 Realizable $k-\varepsilon$ 模型

Realizable $k-\varepsilon$ 模型与标准 $k-\varepsilon$ 模型的主要区别是：

（1）Realizable $k-\varepsilon$ 模型中采用了新的湍流粘度公式。

（2）ε 方程是从涡量扰动量均方根的精确输运方程推导出来的。

现实 $k-\varepsilon$ 模型满足对雷诺应力的约束条件，因此可以在雷诺应力上保持与真实湍流的一致。这一点是标准 $k-\varepsilon$ 模型和 RNG $k-\varepsilon$ 模型都无法做到的。这个特点在计算中的好处是，可以更精确地模拟平面和圆形射流的扩散速度，同时在旋转流计算、带方向压强梯度的边界层计算和分离流计算等问题中，计算结果更符合真实情况。

Realizable $k-\varepsilon$ 模型是新出现的 $k-\varepsilon$ 模型，虽然还无法证明其性能已经超过 RNG $k-\varepsilon$ 模型，但是在分离流计算和带二次流的复杂流动计算中的研究标明，Realizable $k-\varepsilon$ 模型是所有 $k-\varepsilon$ 模型中表现最出色的湍流模型。

Realizable $k-\varepsilon$ 模型在同时存在旋转和静止区的流场计算中，比如多重参考系、旋转滑移网格等计算中，会产生非物理湍流粘性，因此在类似计算中应该慎重选用这种模型。

7.2.6 $k-\omega$ 模型

$k-\omega$ 模型也是二方程模型。标准 $k-\omega$ 模型中包含了低雷诺数影响、可压缩性影响和剪切流扩散，因此适用于尾迹流动计算、混合层计算、射流计算，以及受到壁面限制的流动计算和自由剪切流计算。

剪切应力输运 $k-\omega$ 模型，简称 SST $k-\omega$ 模型，综合了 $k-\omega$ 模型在近壁区计算的优点和 $k-\varepsilon$ 模型在远场计算的优点，将 $k-\omega$ 模型和标准 $k-\varepsilon$ 都乘以一个混合函数后再相加就得到这个模型。在近壁区，混合函数的值等于 1，因此在近壁区等价于 $k-\omega$ 模型。在远离壁面的区域混合函数的值则等于 0，因此自动转换为标准 $k-\varepsilon$ 模型。

与标准 $k-\omega$ 模型相比，SST $k-\omega$ 模型中增加了横向耗散导数项，同时在湍流粘度定义中考虑了湍流剪切应力的输运过程，模型中使用的湍流常数也有所不同。这些特点使得 SST $k-\omega$ 模型的适用范围更广，比如可以用于带逆压梯度的流动计算、翼型计算、跨音速激波计算等等。

7.2.7 v^2-f 模型

v^2-f 模型与 $k-\varepsilon$ 模型比较类似，但是 v^2-f 模型中考虑到了壁面附近湍流的各向异性问题和非局部的压强与应变的关系。v^2-f 模型属于低雷诺数湍流模型，其适用范围从自由流区一直延伸到壁面，并且无需适用壁面函数。v^2-f 模型主要用于边界层计算和分离流计算。

v^2-f 模型的突出特点是用速度尺度 $\overline{v^2}$ 代替湍流动能 k 计算湍流粘度。速度尺度 $\overline{v^2}$ 代表速度的脉动量，因此可以更准确地模拟湍流变量在壁面附近的耗散过程。这点是用湍流动能 k 进行计算时无法做到的。

7.2.8 雷诺应力模型（RSM）

雷诺应力模型中没有采用涡粘度的各向同性假设，因此从理论上说比湍流模式理论要精确得多。雷诺应力模型不采用 Boussinesq 假设，而是直接求解雷诺平均 NS 方程中的雷诺应力项，同时求解耗散率方程，因此在二维问题中需要求解 5 个附加方程，在三维问题

中则需要求解 7 个附加方程。

从理论上说，雷诺应力模型应该比一方程模型和二方程模型的计算精度更高，但实际上雷诺应力模型的精度受限于模型的封闭形式，因此雷诺应力模型在实际应用中并没有在所有的流动问题中都体现出其优势。只有在雷诺应力明显具有各向异性的特点时才必须使用雷诺应力模型，比如龙卷风、燃烧室内流动等带强烈旋转的流动问题。

7.2.9 湍流模型的计算速度

计算速度的快慢与计算量成反比，即计算量大则计算速度慢，需要的时间也长。湍流模型计算中的工作量主要取决于方程的数量和方程中函数项的多少。如果不考虑大涡模拟方法，湍流模型计算从总体上说，一方程模型（Spalart-Allmaras 模型）计算最快，二方程（$k-\varepsilon$ 模型、$k-\omega$ 模型、v^2-f 模型）模型次之，雷诺应力模型最慢。

7.2.10 壁面函数和近壁模型

在受壁面限制的流动中，因为壁面附近流场变量的梯度较大，所以壁面对湍流计算的影响很大。湍流模型中假定湍流是各向同性的，因此在壁面附近需要进行特殊处理。处理的一种办法是用半经验公式将自由流中的湍流与壁面附近的流动连接起来，这种方法被称为壁面函数法。另一种方法是通过在壁面附近加密网格，同时调整湍流模型以包含壁面影响的方法，被称为近壁模型法。

壁面函数法中又有标准壁面函数法和非平衡壁面函数法。一般地说，标准壁面函数可以适用于大多数流动问题，因此也是 FLUENT 中缺省设置的方法。非平衡壁面函数法则适用于流场函数在壁面附近存在很大梯度的流动问题。

壁面函数法适用于高雷诺数流动，近壁模型法适用于低雷诺数流动。

7.2.11 湍流计算方法的设置

湍流计算方法的选择在 Viscous（粘性）面板中进行。在确定了所需要的湍流模型之后，逐项进行选择即可，这里不再重复其操作过程。

7.3 活动变形区域中的流动计算

在计算单一运动物体内部或外部的流场时，可以通过将坐标系定义在物体上的办法，将问题转化为物体静止而来流变化的流动问题。比如在计算搅拌器内的流动时，就可以用

这种办法将参考坐标系连接在搅拌轮上，而将搅拌桶定义成动边界。从这个坐标系看去，搅拌轮是静止的，而搅拌桶是转动的，因而通过坐标转化可以用固定网格完成流动计算。

然而在多个物体之间存在相对运动时，简单地转换参考系显然是不能解决问题的。比如在计算涡轮发动机中的静子与转子的干扰问题时，无论怎样设置参考系，都会遇到固体边界随时间变化的问题。FLUENT 中用于这类问题计算的模型包括多重参考系（Multiple Reference Frame, MRF）模型、混合面（Mixing Plane）模型、滑移网格（Sliding Meshes）技术。如果固体边界不是发生平动或转动，而是随时间变化发生形变，则可以用动态网格（Dynamic Meshes）技术进行计算。

7.3.1 旋转坐标系中的流动计算

采用旋转坐标系进行流动计算时，需要将坐标系与外形复杂的部件连接在一起，而被设置为做旋转运动的部件，在几何上必须是以旋转中心为圆心形成的一个圆或圆柱。这样可以将问题转化为定常流动问题，从而大大简化计算。比如在前面提到的搅拌器中，搅拌桶是一个以搅拌轴为对称轴的圆柱体，搅拌轮则是一个带叶片的复杂几何体。如果将坐标系与搅拌轮相连，则只要在 FLUENT 中将作为边界的搅拌桶壁面设为运动边界，同时将其转动角速度取为搅拌轮角速度的负值，则整个问题就可以被简化为定常问题。反之，如将坐标系与搅拌桶相连接，则会使问题复杂化。

网格设置的注意事项

在采用旋转坐标系进行计算时，在网格使用上 FLUENT 规定：

（1）在二维问题中，旋转轴必须平行于 Z 轴，即垂直于屏幕所在平面。

（2）在轴对称问题中，计算域取为过旋转轴的一个二维平面区域，此时必须将旋转轴设为 X 轴。

（3）在三维问题中，旋转轴可以取为任意直线，但是为方便起见，最好还是将旋转轴设为某个坐标轴。

旋转坐标系的设定过程

采用旋转坐标系计算时，相关设置过程为：

（1）在 Solver（求解器）面板中的 Velocity Formulation（速度公式）项目下选择 Relative（相对速度）或 Absolute（绝对速度）。因为在用耦合求解器时只能使用绝对速度，所以不用设置此项目。

（2）在 Boundary Conditions（边界条件）面板中，从边界列表中选择旋转边界，然后点击 set（设置）按钮打开 Fluid（流体）面板或 Solid（固体）面板进行边界条件设置，包括 Rotation-Axis Origin（旋转轴原点）和 Rotation-Axis Direction（旋转轴方向）的设置，同时 Motion Type（活动类型）中选择 Moving Reference Frame（活动参考系），然后在 Rotational Velocity（旋转速度）下设置 Speed（速度）。

（3）定义壁面速度。

（4）在速度入口处定义速度，在压强入口定义流动方向和总压。

总压和流动方向的定义都取决于速度的定义方式，即如果使用绝对速度定义，总压和流动方向均在绝对坐标系中定义；在使用相对速度定义时，总压和流动方向则相对于旋转坐标系定义。

在压强出口，静压定义与坐标系无关。在回流区，无论采用绝对速度还是相对速度定义，给定的静压值都被用作回流的总压，而流动方向都被假定垂直于压强出口边界。

求解策略

- 在旋转强烈的区域，建议使用相对速度定义。在其余区域，建议使用绝对速度定义。
- 建议使用 PRESTO!格式。
- 在网格划分时，应该在压强、速度梯度较大的区域加密。
- 降低亚松弛因子，比如可以将亚松弛因子调整为 0.3-0.5，甚至更低。
- 开始计算的时候可以从较低的旋转速度算起，然后逐渐增加旋转速度。

7.3.2 多重参考系（MRF）模型

如前所述，在计算域中同时存在动网格区和静网格区时，可以采用的模型包括 MRF 模型、混合面模型和滑移网格模型。其中 MRF 模型和混合面模型用于定常流计算，而 MRF 则是最简单的，因而也是最经济的模型。

在使用 MRF 模型进行计算时，整个计算域被分成多个小的子域。每个子域可以有自己的运动方式，或静止，或旋转，或平移。流场控制方程在每个子域内进行求解，在子域的交界面上则通过将速度换算成绝对速度的形式进行流场信息交换。

如果交界面上网格是正则的，即交界面两侧区域共用相同的网格节点，则网格划分方法与常规方法别无二致。如果网格是非正则的，其处理方法请参阅本书中与非正则网格处理相关的内容。

MRF 模型的设置

与旋转坐标系的设置类似，MRF 模型的设置方法为：

（1）在 Solver（求解器）面板中选择速度的定义形式，即决定采用相对速度定义，还是绝对速度定义。当然在使用耦合求解器时，不用进行此项设置。

（2）在 Boundary Conditions（边界条件）面板中选择相应的边界名称，点击 set（设置）按钮进入 Fluid（流体）面板或 Solid（固体）面板，并设置相应的旋转轴原点和旋转轴方向。在 Motion Type（运动类型）列表中选择 Moving Reference Frame（移动参考系），并设置 Rotational Velocity（旋转速度）下的 Speed（速度），或 Translational Velocity（平动速度）的 X、Y、Z 分量。

（3）在壁面上设置速度。

（4）在速度入口定义速度，在压强入口定义总压和流动方向。

注意事项

（1）最好不要与 Realizable $k-\varepsilon$ 模型配合使用。

（2）活动网格区域在与相邻区域的交界面上的法向速度应该为零。

（3）最好将 MRF 模型的使用限制在定常流计算中，而将非定常流计算交给后面介绍的滑移网格模型进行计算。

（4）在采用相对速度定义时，无质量粒子的迹线显示是有意义的，而有质量粒子，包括弥散相粒子的迹线显示是无意义的。因此在需要跟踪粒子轨迹的时候，最好采用绝对速度定义。

（5）在采用相对速度定义时，MRF 对轴对称旋流的计算不够精确。

（6）MRF 模型中的平移速度和旋转速度都是常数，即边界上速度应该是“均匀”的。

7.3.3 混合面模型

在 MRF 模型的计算中，边界上的速度必须是相同的，否则得出的结果可能是没有意义的。在边界上各点的速度并不相同时，当然可以采用滑移网格法进行计算，但是考虑到滑移网格法消耗的系统资源较大，所以在一些情况下，混合面模型可以作为一个很好的补充，用于进行这类计算。

基本思想

与 MRF 模型一样，混合面模型也将计算域划分为数个彼此独立的子域。与 MRF 模型不同的是，混合面模型在子域的交界面上构造了一个混合面，即用面积平均的方法将交界面两侧的流场变量联系起来，而每个子域内的流场则可以作为定常流场进行求解。

例如，在多级压气机的计算中，压气机由多级静子和转子组成，每级的叶片数量都不一样。在这种情况下，可以将每级静子或转子与相邻流场区域划分出来作为计算的子域，而在交界面上用混合面相连接。这样每个划分出来的子域都可以采用不同的坐标系和边界条件，并转化为定常问题进行计算。子域间的信息交流则通过混合面完成。

混合面模型的计算过程为，首先计算各子域内的定常流场，其次计算混合面上的变量分布，然后以混合面上的变量分布为边界条件开始下一个迭代步的计算，直到计算收敛。

混合面模型的设置

（1）在 Solver（求解器）面板中设定速度的定义形式，即确定采用相对速度还是绝对速度定义。

（2）设定旋转轴原点和方向，同时设定旋转速度。

（3）设定边界上的速度。

（4）设定速度入口和压强入口条件。

（5）在 Mixing Planes（混合面）面板中定义混合面：

Define->Mixing Planes...

- 指定混合面的上、下游计算子域，即在 Upstream Zone（上游区域）和 Downstream Zone（下游区域）中指定上、下游子域的名称。
- 在三维问题中，还需要指定混合面的几何形状，即指定混合面为径向（Radial）平面，还是轴向（Axial）平面。径向平面的含义是，混合面包含旋转轴直线，流量在圆周方向进行平均。轴向平面的含义是，混合面与旋转轴相互垂直，流量沿轴向进行平均。

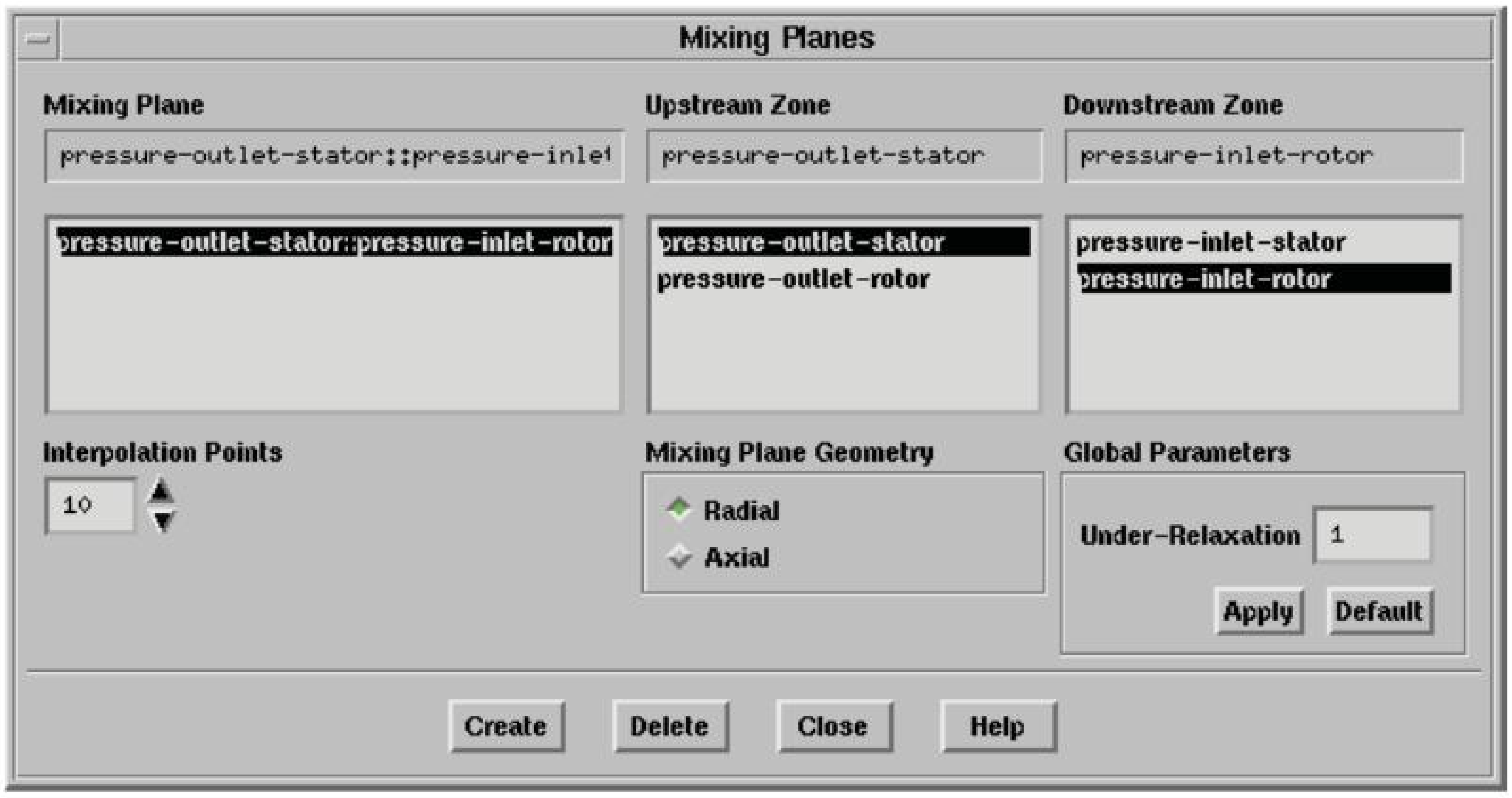


图 7-3 混合面面板

- 设定 Interpolation Points（插值点）的值，这个值是用于构造变量分布曲线的插值点数目。
- 设定 Global Parameters（全局参数），即设定混合面计算的亚松弛因子。
- 点击 Create（创建）按钮，完成混合面的创建过程。

混合面模型的选项

（1）在不可压流计算中，为了控制压强水平，可以用文本命令 fix-pressure-level（固定压强水平）：

define->mixing-planes->set->fix-pressure-level

启用这个选项后，在每次迭代结束后，FLUENT 都会从表压中减去最靠近参考压强位置的网格单元上的压强值。参考压强位置在 Operating Conditions（操作条件）面板中的

Reference Pressure Location（参考压强位置）选项中设置。

（2）用文本命令 conserve-swirl（旋流保持）可以保持涡量在混合面前后的守恒：

define->mixing-planes->set->conserve-swirl

用文本命令 enable?可以启动这项设置。启动后可以让求解器在计算过程中报告与涡量变换的相关信息。如果同时使用命令 verbosity?，FLUENT 会在每次迭代时报告指定区域的涡量，包括涡量在上、下游区域上的积分值，以及进行修正前，上、下游涡量积分的比例。如果想得到各压强入口、速度入口、质量入口的涡量积分报告，可以使用命令 report-swirl-integration（旋流积分报告）。

（3）混合面上可以加入的另一个选项是在混合面前后可以保证总焓守恒。因为对于压气机之类的涡轮机械而言，总体效率是直接与每级叶片前后总焓的改变相关的。保持总焓守恒的方法就是调整下游温度分布函数，以使其积分值与上游相等。

总焓守恒可以用文本命令 conserve-total-enthalpy 进行设定：

define->mixing-planes->set->conserve-total-enthalpy

同样，开启总焓守恒选项也必须使用 enable?命令。开启后，也可以让求解器在计算过程中报告总焓守恒信息。如果开启 verbosity?，FLUENT 将在每个迭代步中报告上、下游的热流通量及上、下游热流通量的比值。

计算策略

混合面方程在出现回流时会出现较大误差，为避免在混合面上产生回流，可以在计算开始阶段采用固定参数进行计算，等流场稳定下来后再恢复正常条件。另外，采用较低的亚松弛因子也对计算的正常进行有帮助。

7.3.4 滑移网格技术

滑移网格技术可以处理非定常问题，这是它与 MRF 模型和混合面模型的最大区别。滑移网格技术处理的通常是带有周期性的问题，比如涡轮机械中转子和静子的相互干扰问题，但是滑移网格也可以计算非周期性问题，比如两列火车交错行驶过程中周围流场的变化。在不需要考虑转子和静子相互干扰的细节时，用 MRF 模型和混合面模型进行计算就可以获得相互干扰的平均效果，但是在需要考虑干扰过程的细节时，则必须使用滑移网格技术。滑移网格技术在计算中需要使用的系统资源比较大，因此在使用滑移网格技术时需要使用配置较高的计算机。

基本思想

在滑移网格计算中，计算域至少包含 2 个以上存在相对运动的子域。每个运动子域至少有一个与相邻子域连接的交界面。原则上交界面形状是任意的，但在实际计算中，交界面的实际形状都设计成在滑移后相邻子域不能相互重叠的形状，或者说交界面上的运动速度必须与交界面相垂直。比如在旋转机械问题中，交界面都设计成轴对称形式，包括圆锥

面、圆柱面等形状。在列车交错问题中，交界面则设计成平面等等。

滑移网格技术中设定的交界面在计算过程中总是有一部分与相邻子域相连，而其余区域则不与相邻子域相连。与相邻子域相连的区域被称为内部区域。与相邻子域不相连的区域，在平动问题中被称为壁面区域，在周期性流动问题中则被称为周期区域。在每次迭代结束后，FLUENT 都会重新计算内部区域的范围，将交界面的其余部分划定为壁面区域或周期性区域，并在壁面区域和周期性区域上则设定相应的边界条件。在新的迭代步中，只计算内部区域上的通量。滑移网格计算中采用非正则网格技术，即交界面两侧子域在交界面上不共用网格节点，因此内部区域不是用交界面两侧的网格面直接构成的，而是通过子域间的相对移动量重新计算得出内部区域的边界位置。

滑移网格技术的设置和求解

滑移网格技术的设置过程如下：

（1）在求解器面板中将计算类型设置为非定常。

（2）在边界条件面板中，将交界面的网格类型设置为 interface。同时在 Fluid 面板和 Solid 面板中，将移动区域的运动类型（Motion Type）设置为移动网格（Moving Mesh），并设定其移动速度。

（3）在 Grid Interfaces 面板上定义网格交界面：

Define->Grid Interfaces...

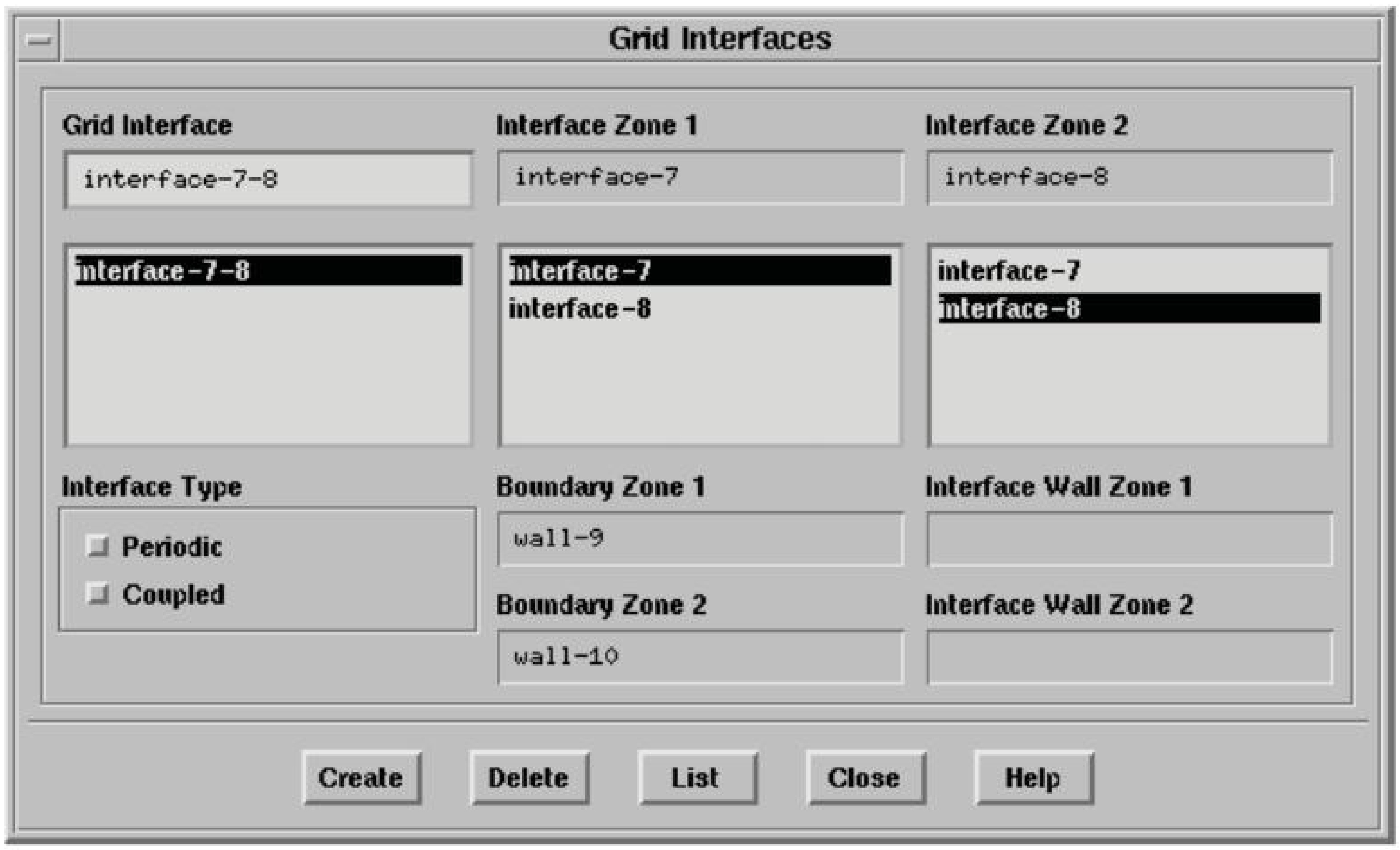


图 7-4 网格交界面

在 Grid Interfaces 中输入交界面的名称，在 Interface Zone 1 和 Interface Zone 2 中指定交界面两侧的流动区域，同时在 Interface Type 中设定交界面类型（周期性问题选 Periodic，

如果交界面位于固体区域和流体区域之间则选 Coupled），最后点击 Create 按钮完成网格交界面的创建。

因为滑移网格技术通常用于非定常问题的计算，所以计算过程中应该注意保存每个时间段的算例文件和数据文件。如果计算的是周期性流动问题，比如旋转机械问题，则在计算开始的时候可以采用大的时间步长以缩短开始阶段的不稳定过程，在计算稳定下来后再减小时间步长以保证时间精度，同时每隔一个周期观察一下流场变量的变化。如果变化逐渐缩小，说明计算是稳定的。在变化量小于 5%时，可以认为计算已经收敛。

在采用二阶时间精度进行计算的过程中，开始阶段可以采用大的时间步长，但是在随后减小时间步长的过程中，每次减小不要超过 20%。在计算的最后阶段，最好不要再改变时间步长，因为那样会严重影响计算的时间精度。

在后处理过程中，速度场的显示在缺省设置中显示的是绝对速度，也可以根据需要将其改为显示相对速度。

7.3.5 动网格技术

动网格技术用于计算运动边界问题。在前面的计算中，通常计算域的边界都是静止的，或做刚体运动的，而动网格技术则可以计算边界发生形变的问题。边界的形变过程可以是已知的，也可以是取决于内部流场变化的。在计算之前首先要给定体网格的初始定义。在边界发生形变后，其内部网格的重新划分是在 FLUENT 内部自动完成的。而边界的形变过程即可以用边界函数来定义，也可以用 UDF 函数来定义。

如果计算域中同时存在运动区域和静止区域，因此在初始网格中，内部网格面或区域需要被归入其中一个类别，同时在运动过程中发生形变的部分也可以单独分区。区与区之间即可以采用正则网格，也可以采用非正则网格，还可以用滑移网格技术连接各网格区域。

基本思路

动网格的计算方法有 3 种，即弹性光顺法（spring-base smoothing）、动态层技术（dynamic layering）和局部网格重划法（local remeshing）。

弹性光顺法将网格系统看作由节点之间用弹簧相互连接的网络系统，初始网格就是系统保持平衡的弹簧网络系统。任意一个网格节点的位移都会导致与之相连接的弹簧中产生弹性力，进而导致临近网格点上的力平衡被打破。由此波及出去，经过反复迭代最终整个网格系统达到新的平衡时，就可以得到一个变形后的、新的网格系统。

动态层技术是根据边界的移动量动态地增加或减少边界上网格层的技术，因此动态层技术适用于六面体网格、楔形网格等可以在边界上分层的网格系统。动态层技术在边界上假定一个优化的网格层高度，在边界发生移动、变形时，如果临近边界的一层网格的高度同优化高度相比大到一定程度时，就在边界面与相邻网格层之间增加一层网格。相反，如果边界向内移动，临近网格被压缩到一定程度时，临近一层网格又会被删除。用这种办法保持边界上的网格保持一定的密度。

局部网格重划法是对弹性光顺法的补充。在网格系统用三角形或四面体网格组成时，如果边界的移动和变形过大，可能导致局部网格发生严重畸变，甚至出现体积为负的情况。在这种情况下，一个简单的处理方法就是去掉由原来网格系统经过弹性光顺得到的新网格，在原来的位置上重新划分网格，这就是局部网格重划法的基本思路。

基本设置

（1）在求解器面板中选择非定常计算。
（2）在边界条件面板中设置动壁面条件。
（3）在动网格模型中设置相关参数：
Define->Dynamics Mesh->Parameters...

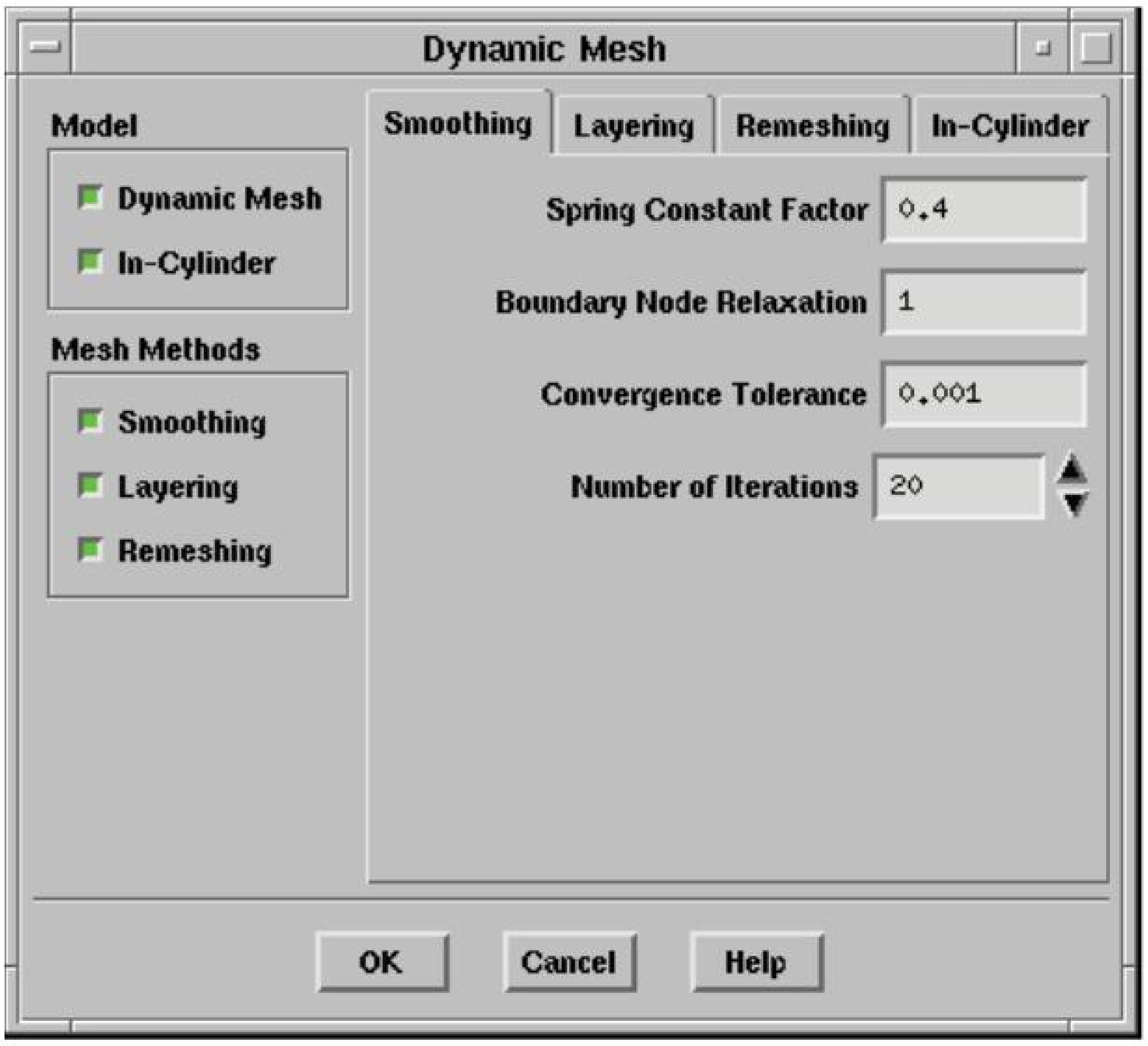


图 7-5 动网格参数设置

汽缸内流动是一类典型的边界移动问题，因此被 FLUENT 单独作为一类问题进行处理。在动网格参数设置中，首先需要选择的就是 Model 中的是否采用动网格（Dynamic Mesh），及所计算的模型是否属于汽缸内的流动（In-Cylinder）。其次就是选择网格的划分方法，包括前面所述的 3 种方法。

在选定网格划分方法后需要在右边的参数标签中选择需要设定的参数，即设定弹性光顺法（Smoothing）、动态层技术（Layering）和局部网格重划法（Remeshing）中需要设定的参数。如果所计算的问题是汽缸内流动（In-Cylinder），还需要为这种流动进行专门的设

定。

在弹性光顺法的参数中，弹性因子（Spring Constant Factor）是介于 0 到 1 之间的一个常数，其中 0 表示弹簧没有阻尼作用。边界节点松弛（Boundary Node Relaxation）则是一个类似于亚松弛因子的参数，每次迭代后新的网格坐标值都等于原坐标加上边界节点松弛与坐标增量的积，缺省设置为 1。其余两个参数则是网格迭代计算的控制参数，一个是容差（Convergence Tolerance），另一个是迭代次数（Number of Iterations）。

在动态层技术中，可以设置的参数包括常数高度（Constant Height），或常数比（Constant Ratio）。分裂因子（Split Factor）和消灭因子（Collapse Factor）是决定在网格变形到什么程度时，需要增加或消除网格层的控制因子。

局部网格重划中最大网格畸变（Maximum Cell Skewness）、最小网格体积（Minimum Cell Volume）和最大网格体积（Maximum Cell Volume）是决定哪些网格需要被集中起来重新划分的依据。畸变率大的，体积过小的和过大的都可以集中起来，进行重新划分。重新划分间隔（Size Remesh Interval）决定新网格的大小，“必须改进的畸变（Must Improve Skewness）”选项则决定是否需要在网格达到要求后再停止重划。

汽缸内流动则通过设定曲柄轴速度（Crank Shaft Speed）、曲柄轴起始角度（Starting Crank Angle）和曲柄轴周期（Crank Period）等定义曲柄周期运动。还有一个需要设定的参数是曲柄角度步长（Crank Angle Step Size），这个参数用于确定曲柄角度的计算步长。FLUENT 同时还提供一个内建的函数用于计算曲柄位置，即活塞顶端位置（Piston Stroke）和连接杆长度（Connecting Rod Length）。

（4）定义动态区域的运动：

Define->Dynamic Mesh->Zones...

首先选择区域名称（Zone Names），然后选择运动类型（Type），包括静止（Stationary）、刚体运动（Rigid Body）、变形（Deforming）和用户定义（User-Defined）等几种。然后设定运动相关属性，即在标签 Motion Attributes 下面根据计算模型设定重心位置等参数。

（5）保存算例文件和数据文件。

（6）预览网格设置：

Solve->Mesh Motion...

首先保存算例文件。然后设定时间步数量（Number of Time Steps）和时间步长（Time Step Size）。当前时间显示在当前网格时间（Current Mesh Time）中。如果计算的是汽缸内流动，则时间步长是用曲柄角度步长（Crank Angle Step Size）和曲柄速度（Crank Shaft Speed）计算出来的。如果要显示动态网格，打开显示网格（Display Grid）选项，并设定显示频率（Display Frequency）。如果需要将显示内容保存在文件中，可以打开保存硬拷贝（Save Hardcopy）选项。点击 Preview 按钮开始预览。

（7）如果求解的是汽缸内的流动问题，则定义与计算相关的事件参数：

Define->Dynamic Mesh->Events...

这个选项仅在选择使用了 In-Clinder（汽缸内流动）模型时可以被打开。首先增加 Number of Events（事件数量），然后在 On（打开）下面点击相应的开关，并输入事件名称，和曲柄角度，最后点击 Define（定义）按钮进入 Define Event（定义事件）面板。根据具体

情况进行适当的设置后，点击 OK 按钮保存并推出。为了确定设置效果，可以在 Dynamic Mesh Events（动态网格事件）面板中点击下面的 Preview（预览）按钮进行观察。如有错误，可以重新进行设置。

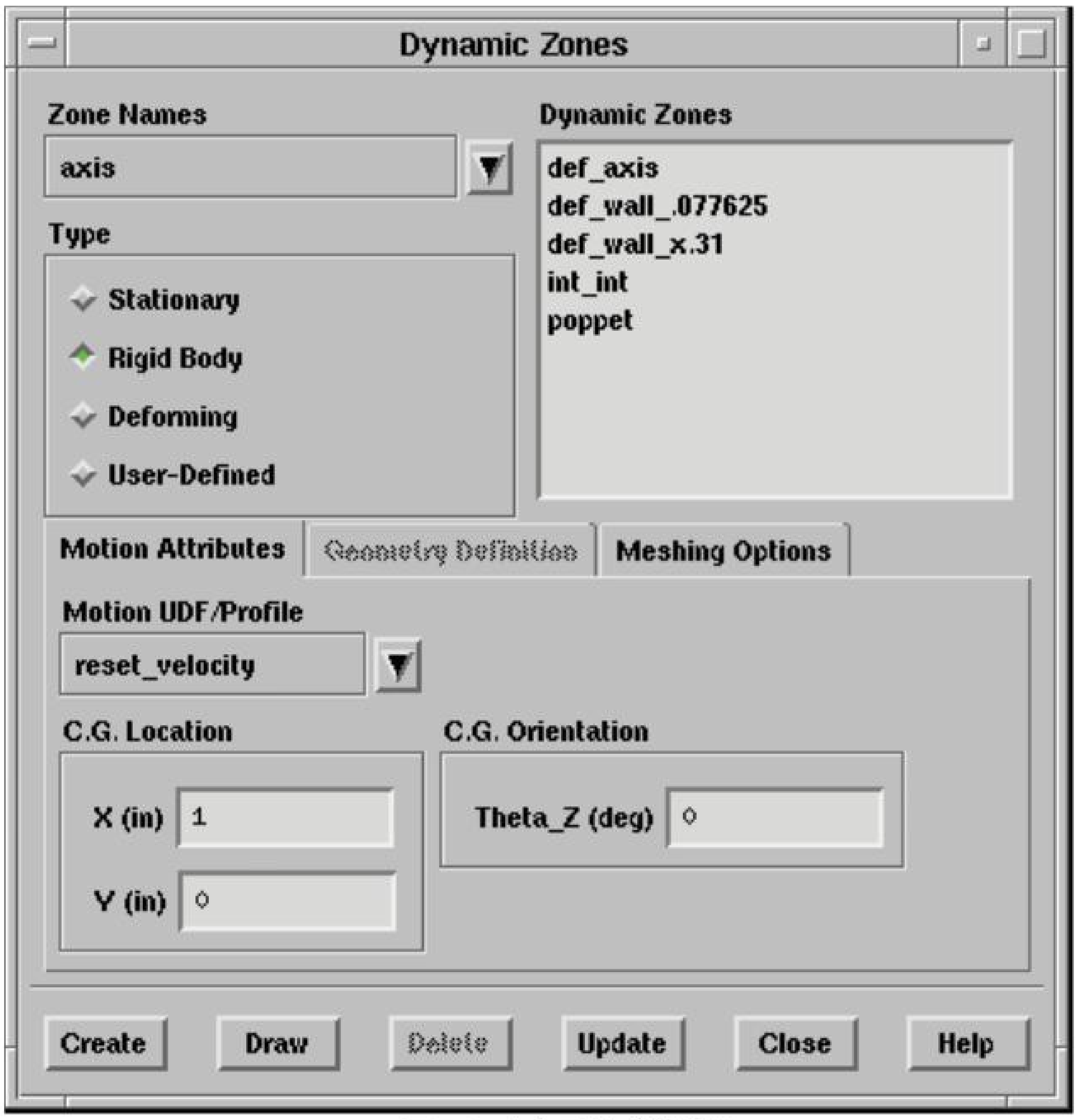


图 7-6 动态区域参数设置

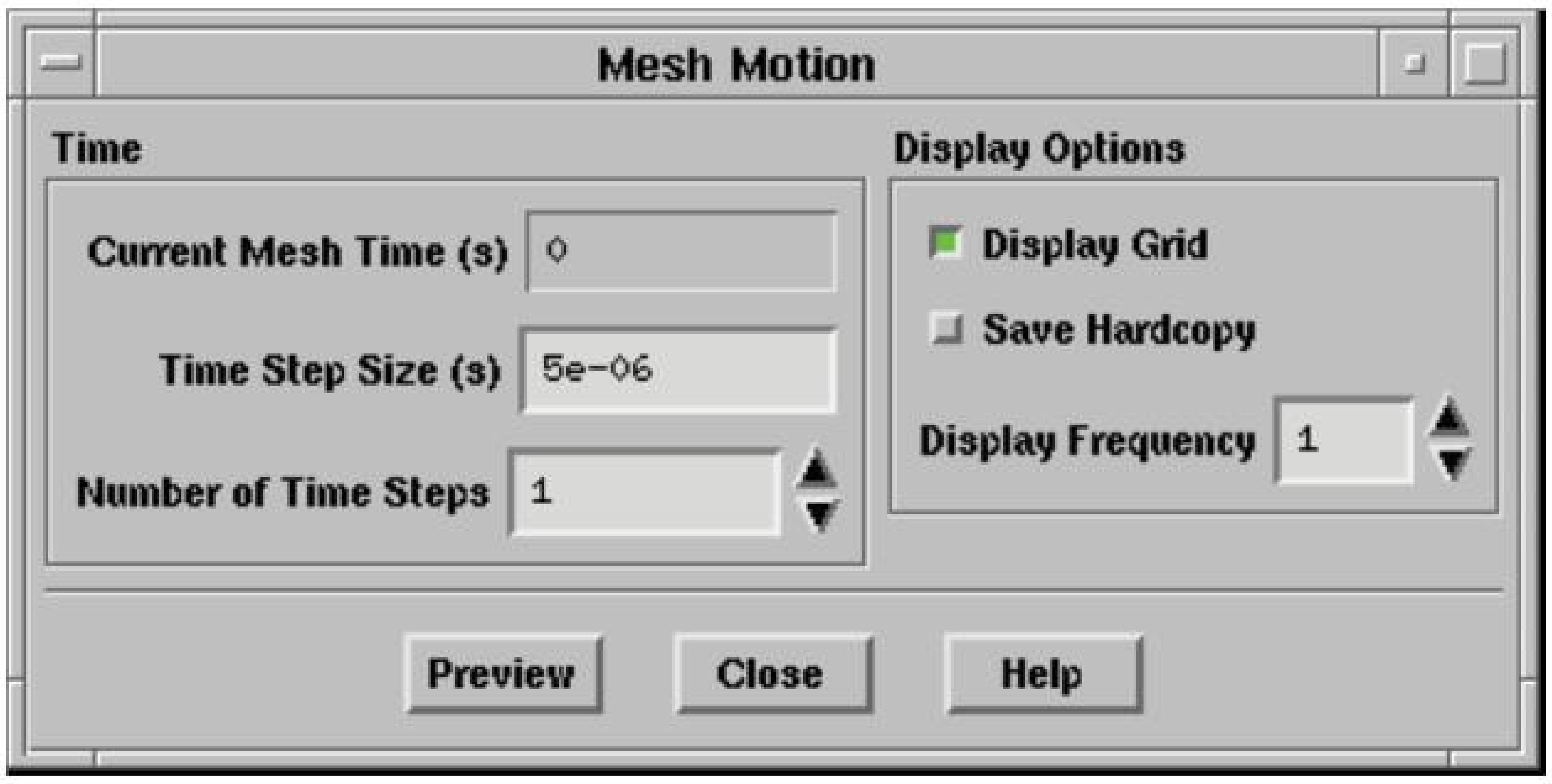


图 7-7 网格运动

除了在 Dynamic Mesh Events 面板中使用 Preview 按钮，还可以用下面的菜单操作显示汽缸内活塞和阀门的运动：

Display->IC Zone Motion...

（8）设置自动保存，以便隔一定的计算步保存一次算例文件和数据文件。

（9）计算中可以启动动画录制过程。

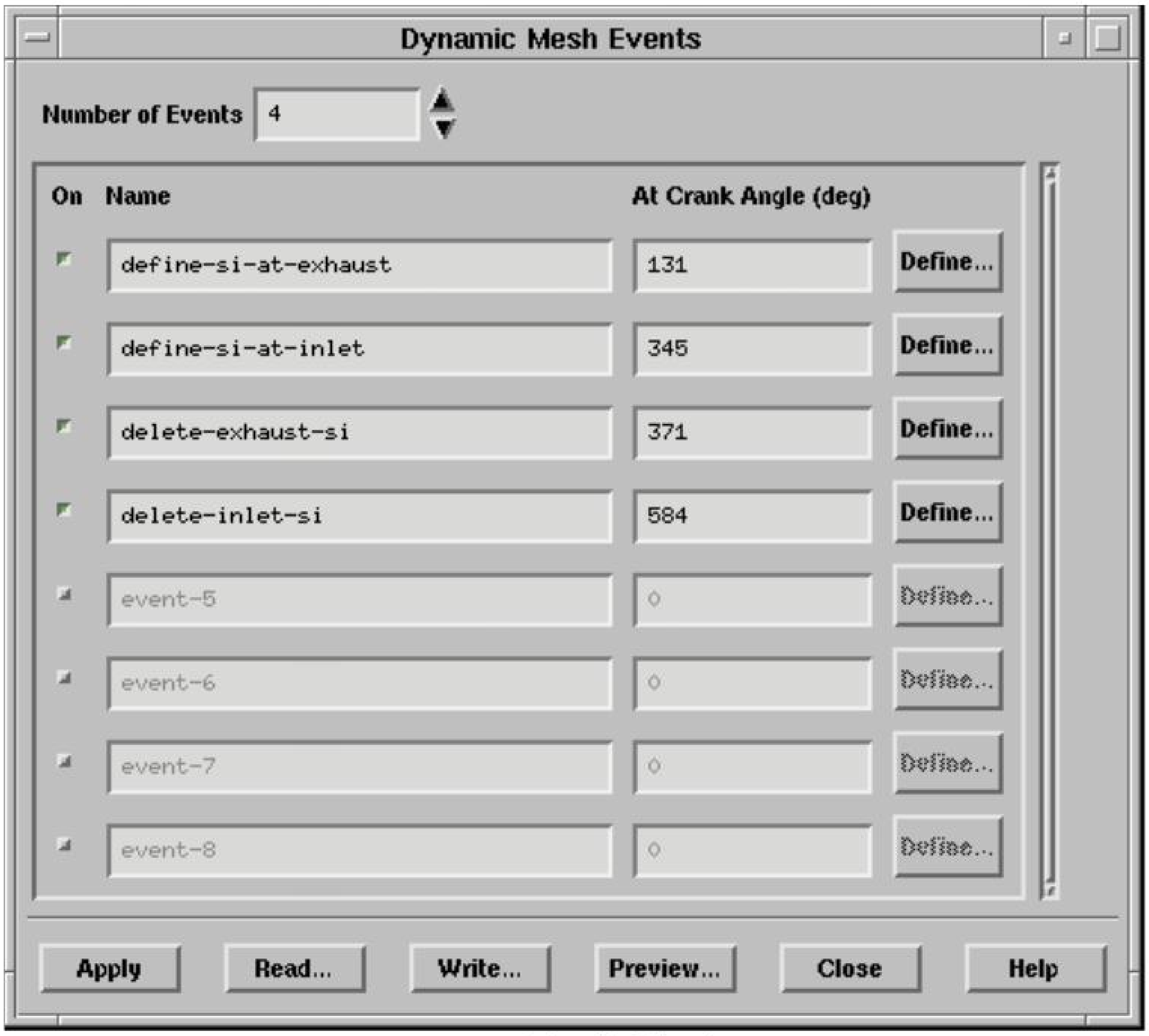


图 7-8 动态网格事件面板

7.4 化学反应模型

FLUENT 可以计算化学组分输运和化学反应过程。化学反应过程从类型上看可以分为气相反应、物面反应和弥散相粒子表面的反应。在气相反应中可以采用的计算方法又包括：

- □ 有限速率化学反应模型。
- □ 非预混燃烧模型。
- □ 预混燃烧模型。

□ 部分预混燃烧模型。

□ PDF 输运燃烧模型。

选择反应模型的一般方法为：

（1）对于涉及到化学反应、混合和输运过程的，或者反应发生在壁面或弥散相粒子表面的，采用有限速率化学反应模型。

（2）在计算湍流火焰扩散问题时，如果燃料和氧化剂是从多个不同入口进入流场，则可以使用非预混模型。

（3）对于所有反应物都均匀混合在一起的情况，可以使用预混燃烧模型。

（4）在火焰为预混火焰，而等价比（equivalence ratio）变化的问题，应该采用部分预混燃烧模型。

（5）如果在湍流火焰的计算中必须考虑有限速率化学反应的话，可以使用层流火苗模型，或组合物 PDF 输运模型。

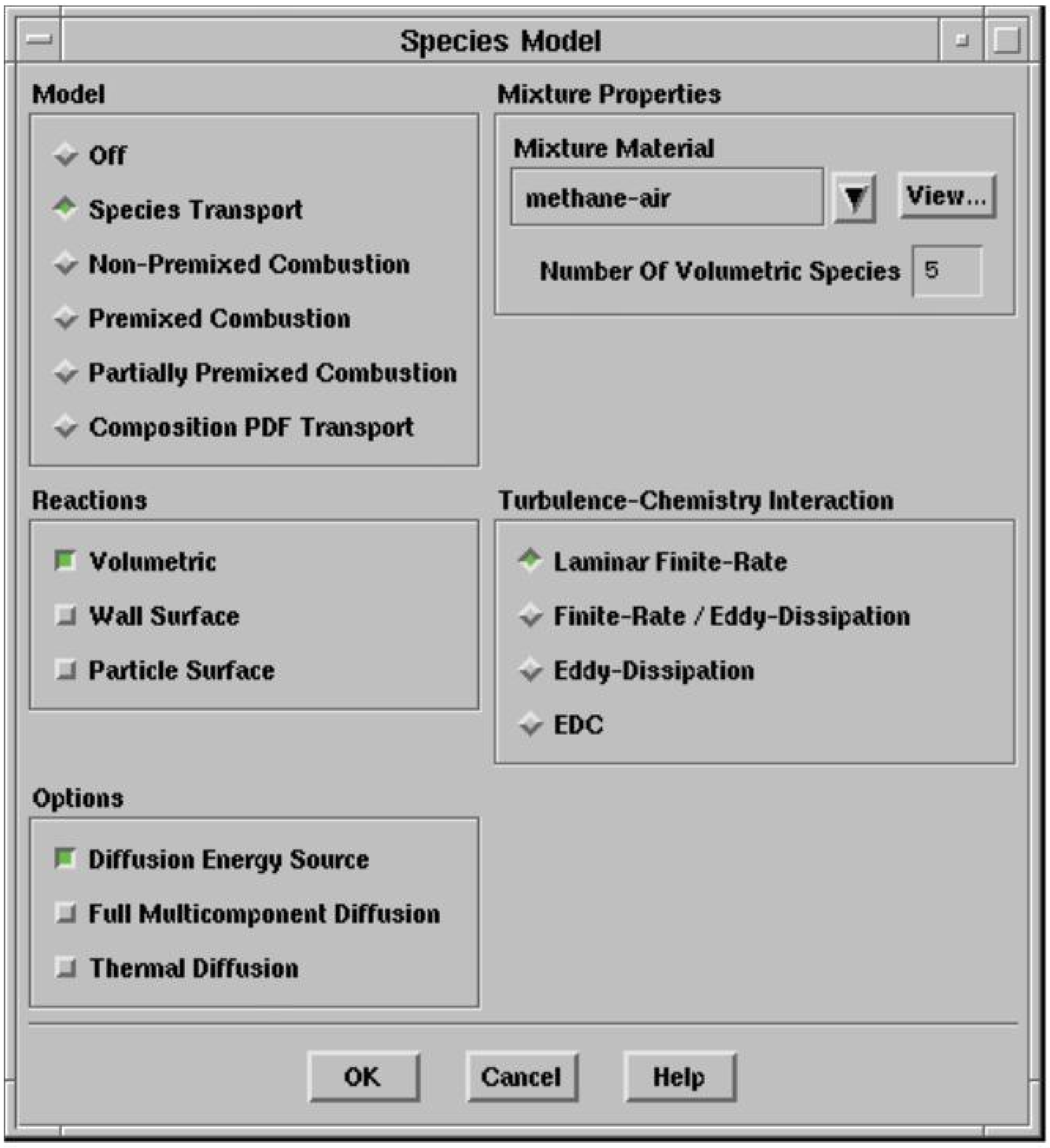


图 7-9 组元模型面板

7.4.1 气相反应

本小节只介绍有限速率化学反应，与燃烧和 PDF 有关的内容请参阅后面的 7.5 和 7.6 两节。有限速率化学反应模型的设置过程如下：

首先启动 Species Model（组元模型）面板，如图 7-9 所示：

Define->Modals->Species...

（1）在 Model（模型）中选择 Species Transport（组元输运）。

（2）在 Reactions（反应）中选择 Volumetric Reactions（体积反应）。

（3）在 Mixture Material（混合物材料）中选择所计算问题中涉及到的反应物，则 Number of Volumetric Species（体积组元数量）中自动显示混合物中的组元数量。

（4）在 Turbulence-Chemistry Interaction（湍流与化学反应相干模型）中根据需要选择相应的模型。

（5）如果选择了 EDC，则可以进一步修改 Volume Fraction Constant（体积浓度常数）和 Time Scale Constant（时间尺度常数）

（6）如果想完整计算多组分的扩散或热扩散，就选中 Full Multicomponent Diffusion（完整多组分扩散）和 Thermal Diffusion（热扩散）选项。

在上面的设置过程中，如果需要查看混合物中组分和化学反应的相关设置，可以在 Species（组元）面板中，点击 Mixture Material（混合物材料）右边的 View（观看）按钮。如果计算中用到的混合物模型是一种新的混合物，则需要在 Material（材料）面板中创建混合物，然后再将新定义的混合物选作计算用的混合物。

混合物的定义过程包含组分选取、反应模型设定、反应机制设定等几个步骤，下面逐一介绍。

组分选取

首先进入 Materials（材料）面板：

Define->Materials...

在 Materials（材料）面板上，先在 Name（名称）中为新的混合物确定一个名称，然后在 Material Type（材料类型）里选择 mixture（混合物）。如果有与目标相近的混合物模型，可以在下面的 Mixture Material（混合物材料）中选择一样，比如 methane-air（甲烷－空气），然后在下面 Properties（性能）中做详细设置，即按顺序设置组元、反应类型、反应机制等等：

（1）点击 Mixture Species（混合物组元）右边的 Edit（编辑）按钮进入 Species（组元）面板，如图 7-11 所示。在 Mixture（混合物）下面有 4 个框，即 Available Materials（可用材料），Selected Species（已选组元），Selected Site Species（已选吸收组元）和 Seleted Solid Species（已选固体组元）。

Available Materials（可用材料）是指材料数据库中可供选用的材料；Selected Species（已选组元）是指当前混合物中已经选中的组元；Selected Site Species（已选吸收组元）是

指在存在物面反应的计算中，气相混合物中即将通过反应被物面吸收的组元；Selected Solid Species（已选固体组元）是指物面反应计算中将从物面进入气流的组元。显然，如果不存在物面反应，则不用考虑后面两个方框中的内容，整个设置过程会大大简化。

组元设置的中心任务是选择混合物组元，即设定 Selected Species（已选组元）的内容。在 Materials（材料）面板中，点击 Database...（数据库）按钮打开 Database Materials（数据库中材料）面板，拷贝所需的组元后，再回到 Species（组元）面板。在拷贝之前，需要确认 Material Type（材料类型）必须是 fluid（流体）。这里不必担心如何添加物面反应中参与反应的固体组元，因为实际上这些组元也会出现在 fluid（流体）的列表中。

在 Selected Species（已选组元）中，最后一项必须是质量浓度最大的一个组元。如果最后一项不是质量浓度最大的一项，可以先将质量浓度最大的组元从方框中删除，然后再重新添加进来，以保证这个组元处于方框的底部。添加和删除操作是通过选择相应的组元，然后点击 Add（添加）和 Remove（删除）按钮实现的。

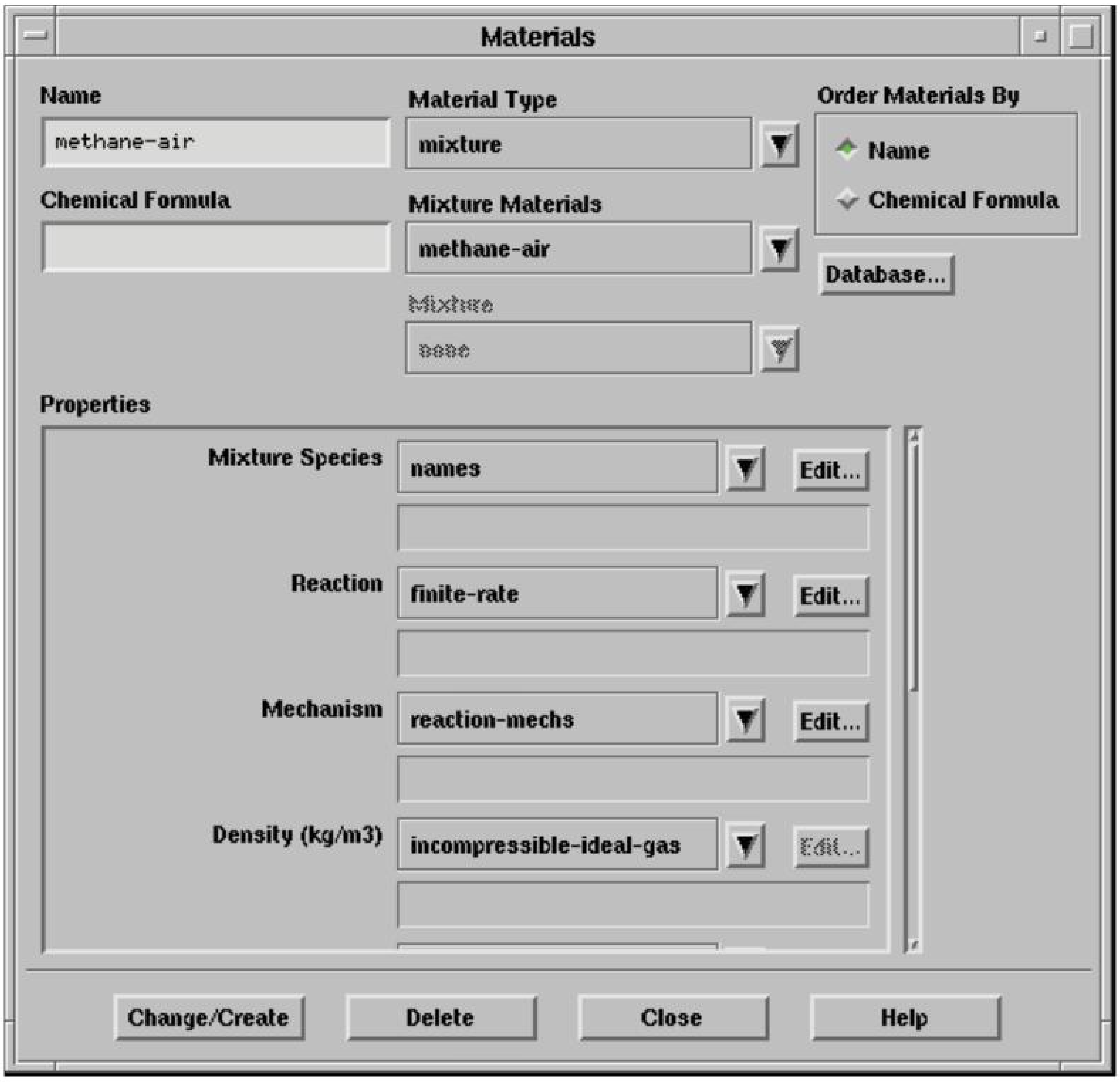


图 7-10 材料面板

化学反应设定

组元设置完成后，就可以开始设置组元间的化学反应。在 Materials（材料）面板中，Reactions（反应）下拉列表中显示的反应类型取决于 Species Modal（组元模型）面板中 Turbulence-Chemistry Interaction（湍流－化学反应相干）模型的设置——如果设置的是 Laminar Finite-Rate（层流有限速率）模型或 EDC 模型，则反应类型显示为 finite-rate（有限速率）；如果设置的是 Eddy-Dissipation（涡耗散）模型，则反应类型显示为 eddy-dissipation（涡耗散）；如果设置的是 Finite-Rate/Eddy-Dissipation（有限速率/涡耗散）模型，则反应类型显示为 finite-rate/eddy-dissipation（有限速率/涡耗散）。点击右端的 Edit（编辑）按钮进入 Reactions（反应）面板（如图 7-13 所示），在 Reactions（反应）面板中完成对化学反应模型的设置。



图 7-11 Species（组元）面板

化学反应的设置主要包括下列几项内容：

1）在 Total Number of Reactions（总的反应数量）中设定总的反应数量；

2）在 Reaction Name（反应名称）中指定反应名称；

3）在 Reaction ID（反应编号）中指定每个反应的编号；

4）在 Reaction Type（反应类型）中指定反应类型，即指定反应类型为 Volumetric（体

积)、Wall Surface（物面）或 Particle Surface（颗粒表面）;

5）在 Number of Reactants（反应物数量）和 Number of Products（生成物数量）中指定反应的反应物数量和生成物数量。

然后在 Species（组元）下拉菜单中选择反应物和生成物，并在 Stoich. Coefficient（反应系数）中设定组元在指定反应方程中的系数。在 Rate Exponent（速率指数）中设定组元的速率指数，即组元生成速率方程中指定反应方程中摩尔浓度项的指数。

6）如果在 Species Modal（组元模型）面板的 Turbulence-Chemistry Interaction（湍流－反应相干模型）中选择的是 laminar finite-rate（层流有限速率）、finite-rate/eddy-dissipation（有限速率/涡扩散）或 EDC 模型时，则需要根据反应模型设置 Arrhenius Rate（Arrhenius 速率）下面的选项。由于篇幅所限，这里不再详述。

7）根据实际反应过程，确定是否选择 Include Backward Reaction（包含逆向反应）选项。

8）如果使用 eddy-dissipation（涡扩散）或 finite-rate/eddy-dissipation（有限速率/涡扩散）模型，则还需要设定 Mixing Rate（混合速率）。

9）重复 2）～8）步直到设置完所有反应，然后点击 OK 按钮完成全部设定过程。

在区域上定义反应机制

FLUENT 中提到的“反应机制（reaction mechanisms)”指的是局限在特定区域中的化学反应。“反应机制”中涉及的反应是前面设定的化学反应的子集。在 Materials（材料）面板中，点击 Mechanisms（机制）旁边的 Edit（编辑）按钮，可以打开 Reaction Mechanisms（反应机制）面板，如图 7-14 所示。具体步骤如下：

1、在 Number of Mechanisms（反应机制数量）中设定反应的数量。

2、设定 Mechanisms ID（反应机制编号）。

3、设定 Name（名称）。

4、在 Reaction Type（反应类型）中设定反应类型。在反应类型确定后，属于这个类型的反应就会出现在 Reactions（反应）列表中。

5、选定反应机制中包含的反应。如果选定的反应类型是 Wall Surface（壁面反应）并且其中包含吸收反应，则还需要对吸收反应进行专门的设定，即设定 Number of Sites（吸收反应数量）、Site Name（吸收反应名称）、Site Density（吸收密度），点击 Define（定义）按钮还可以进一步选定吸收反应中的被吸收组元和被吸收组元的 Initial Site Coverage（初始吸收覆盖率）等参数。

设定混合物性质和组元性质

在上述设置结束后，可以分别设定混合物的物理性质，即密度、粘度、热导率和质量扩散系数等参数。同时还需要设定组元的理化性质，包括分子量、生成焓、粘度、热导率、比热和参考温度等参数。

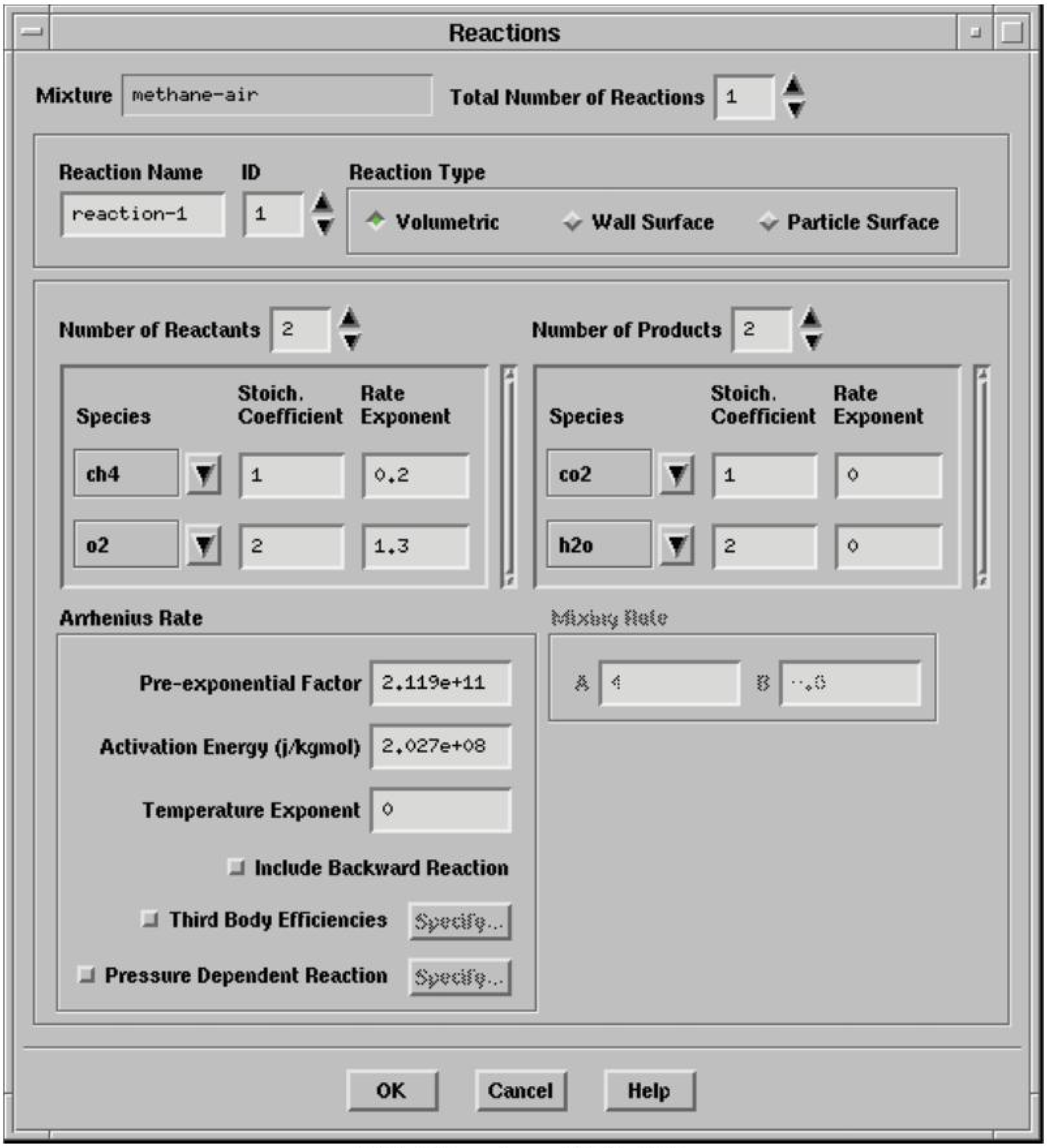


图 7-12 Reactions（反应）面板

边界条件

在有化学反应的计算中，需要在边界上设定组元的质量浓度。在没有壁面反应的时候，通常将壁面上的质量浓度梯度设为零。另外，还可以在 Fluid（流体）面板中为某种组元设置源项。FLUENT 还可以直接从 CHEMKIN 中输入反应模型，菜单操作为：

File->Import->CHEMKIN Mechanism...

指定了相关文件的路径并为反应命名后，就可以直接从 CHEMKIN 中读入反应模型了。

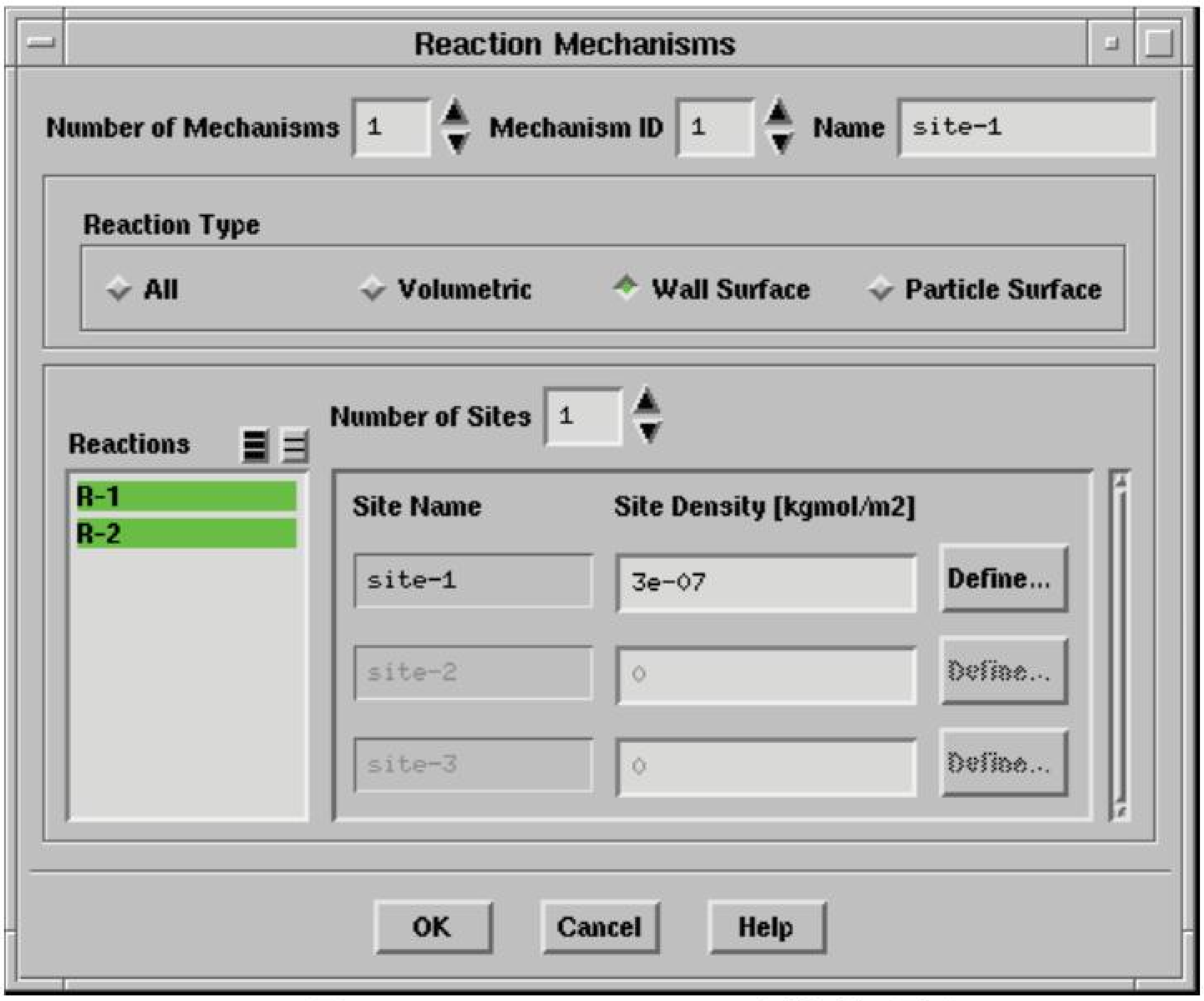


图 7-13 Reaction Mechanisms（反应机制）面板

7.4.2 壁面反应

壁面反应是发生在壁面及附近区域的化学反应。同前面的气相反应相比，明显的一个不同是，壁面反应中有些组元不是通过参与其他反应转变为其他组元，而是通过反应沉积在物面上，因此在组元方程中对源项的物理模型是不同的。同时壁面反应是以单位面积组元的生成率作为基本度量标准的，而在气相反应中是以单位体积来进行度量的。虽然有这些不同，在具体操作上是基本类似的：

（1）首先设定组元模型。在 Species Model（组元模型）面板上进行下列操作：

1）在 Model（模型）下面选择 Species Transport（组元输运）选项。

2）在 Reactions（反应）下面选择 Volumetric（体积）和 Wall Surface（壁面）选项。

3）在 Mixture Properties（混合物性质）下选择 Mixture Material（混合物材料）。

4）如果需要模拟壁面反应传热过程，则开启 Heat of Surface Reactions（表面反应热）。

5）如果需要模拟传质过程，则开启 Mass Deposition Source（质量沉积源项）。

6）如果计算使用的是分离算法，并且不希望在能量计算中考虑组元扩散效应，则关闭 Diffusion Energy Source（扩散能量源项）。

7）如果需要全面模拟多组分扩散火热扩散过程，开启 Full Multicomponent Diffusion（全

面多组分扩散）或 Thermal Diffusion（热扩散）选项。

（2）定义混合物性质。如同在气相反应中那样设定混合物中包含的组元、反应和组元性质等参数。壁面反应中可能被壁面吸收的组元也会出现在 Fluid Material（流体材料）列表中，因此将流体类型设定为 fluid（流体）就可以在列表中找到所有参与反应的组元。

（3）检查混合物中每个组元的性质，如果性质参数不符合反应模型的要求，可以进行组元性质的重新设定。

（4）设定组元边界条件。如果需要考虑壁面反应对壁面的影响，可以在 Wall（壁面）面板中 Species（组元）框内，开启 Surface Reactions（壁面反应）选项。在需要特殊处理的壁面附近设定壁面区附近的“反应机制”。

7.4.3 粒子表面反应

模拟弥散相颗粒表面发生的化学反应是燃烧计算中必不可少的内容。

计算模型设置

在计算中的设置过程如下：

（1）在 Species Model（组元模型）面板中的 Reactions（反应）框下面，开启 Particle Surface（粒子表面反应）选项。

（2）指定参与粒子表面反应的所有组元，包括参与沉积过程的组元。

（3）设定表面反应，即在 Reactions（反应）面板中，Reaction Type（反应类型）下设定反应类型为 Particle Surface（粒子表面反应），并指定下列参数：

1）Diffusion Limited Species（扩散限制组元），即在粒子表面和气流间梯度最大的组元。在只存在一种气相反应物时，没有必要设置这个参数。

2）Catalyst Species（催化组元）。在反应物中不包含固体颗粒时，固体颗粒是被作为催化剂加以定义的，即指定固体颗粒为催化组元。

3）Diffusion Rate Constant（扩散率常数）。

4）Effectiveness Factor（效率因子）。

采用多重表面反应模型

如果在颗粒表面反应中采用多重表面反应模型，其设置过程如下：

（1）启用所有弥散相模型选项。

（2）定义初始条件。

（3）定义边界条件。

（4）定义材料性质。

（5）如果参与表面反应的组元数大于 1，则回到 Set Injection Properties（设置引射性质）面板，或 Set Multiple Injection Properties（设置多重引射性质）面板，定义表面反应组元的质量浓度。

（6）设置求解参数并开始求解。

（7）检查结果。

上面各项的设置过程细节请参考本章后面关于弥散相计算的相关内容。

7.4.4 组元输运

除了进行气相反应计算、颗粒表面反应计算外，FLUENT 还可以进行组元混合问题计算，即在不发生化学反应的情况下计算组元之间的相互参混过程。这一计算过程可以看作是对气相反应计算的简化，基本步骤如下，详细过程请参考前面的反应流计算：

（1）在 Species Model（组元模型）面板中打开 Species Transport（组元输运）选项。

（2）如果需要计算多重扩散和热扩散，则开启 Full Multicomponent Diffusion（全面多重扩散）选项或 Thermal Diffusion（热扩散）选项。

（3）定义混合物及组元性质，即定义混合物中的组元构成和相应的物理性质。

（4）设置组元边界条件。

（5）设置计算条件。

（6）开始计算。

7.5 燃烧模型

燃烧计算分为非预混燃烧、预混燃烧和部分预混燃烧三种，本小节将针对三种算法分别进行介绍。

7.5.1 非预混燃烧模型

非预混燃烧适用于燃料和氧化剂分别来自不同入口的情况，也就是说燃料和氧化剂在燃烧前没有进行过混合，这就是所谓“非预混”的含义。

非预混燃烧计算中不使用有限速率化学反应模型，而是用统一的混合物浓度作为未知变量进行求解，因此计算中无需计算代表组元生成或消失的源项，计算速度比有限速率化学反应模型要快。但是非预混燃烧计算需要流场满足一定的条件，即流场必须为湍流，化学反应过程的驰豫时间非常短，燃料、氧化剂必须来自不同的入口等等。在燃料入口和氧化剂入口之外，非预混燃烧允许存在第三个流动入口，这个入口可以是燃料、氧化剂，也可以是不参与燃烧反应的第三种流体的入口。

非预混燃烧计算使用的化学反应模型包括火焰层近似（flame sheet approximation）、平衡流计算和层流火苗（flamelet）模型三种模型。火焰层近似模型假设燃料和氧化剂在相遇后立刻燃烧完毕，即反应速度为无穷大，其好处是计算速度快，缺点是计算误差较大，特别是对于局部热量的计算可能超过实际值。平衡流计算是用吉布斯自由能极小化的方法求